When Mike started writing poetry I didn't know whether to laugh or cry. He wanted to publish it! Shows how thoroughly humanity had corrupted this innocent machine that he should wish to see his name in print.

– Robert A. Heinlein, *The Moon is a Harsh Mistress*

University of Alberta

A FRAMEWORK FOR EMOTIONAL INFLUENCES ON PROBLEM SOLVING BY SYNTHETIC AGENTS

by

John Arnold ©

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science**.

Department of Computing Science

Edmonton, Alberta
Fall 2006

Your file Votre référence
ISBN: 978-0-494-22224-9
Our file Notre référence
ISBN: 978-0-494-22224-9

Abstract

We develop a framework for emotional modelling that is based on a plan selection architecture. The emotional module determines emotional state based on goal and plan outcomes. The first novel aspect of the framework is that the current emotional state in turn influences the expected utility computations of untried plans, affecting performance on the core problem solving task. A second novel aspect is the inclusion of plan-based behaviours that are triggered by the simulated emotional state and cause that state to dissipate. These behaviours compete for execution with the core problem solving plans. On abstract planning problems, agents modelled with these two emotion framework elements exhibited clearly different problem solving choices than agents modelled without them. Using narrative traces of concrete problem solving behaviour, a study with human subjects found that agents modelled with the emotional module are perceived differently than those that are not, on some personality traits.

Acknowledgements

None of this work would be possible were it not for the invaluable guidance and support of my supervisor, Renée Elio. Thank you very much for all your help during my studies.

Mike Carbonaro and Rob Holte, as part of my defense committee, provided a number of insightful comments and suggestions to improve the work. Thanks to you for reading the whole thing (!) and helping to make it a success.

This work was originally inspired by the idea of fusing computing science and theatre. Gerd Hauck and Andy Houston opened my eyes to the world of theatre at the University of Waterloo. Here in Alberta, Ben Unterman and Piet Defraeye were a wonderful source of ideas and feedback for the project. Thanks to all of you for lending your theatrical perspective.

Thanks to Daniel Torres and Pierre Boulanger for providing ideas, resources, and support for the explorations I did concerning the graphical realization of synthetic agents.

I was lucky to be around a great group of graduate students in the Department of Computing Science, many of whom were always happy to discuss a problem, suggest an idea, or throw a frisbee. Thanks to you all, and special thanks to Peter McCracken, Dan Lizotte, and Brian Tanner for your help at various times.

Writing a thesis while working full time is not an strategy I recommend. However, that's what happened, and I thank my colleagues at CodeBaby for fostering a great working environment and for having patience while I completed my research.

Thanks to all of the staff at Muddy Waters for providing a friendly place to work on my thesis (which is made primarily of lattes and lemon cookies).

A big thanks to all my friends and family for helping to keep me sane through all this! There are far too many people to list here, but you all know who you are.

Finally, I thank Mom & Dad for always being there for me, throughout my education and beyond.

To my superlative friends,
from Vancouver to Reykjavík and all points in between,
for all the encouragement and good times.

Table of Contents

1 Introduction **1**
1.1 Motivation . . . 1
1.2 Approach . . . 1
1.3 Evaluation . . . 2
1.4 Thesis Organization . . . 3

2 Literature Review **4**
2.1 Opening Remarks . . . 4
2.2 Psychological Theories and Models . . . 8
2.2.1 OCC Model . . . 8
2.2.2 OCEAN Personality Model . . . 19
2.3 Implementations of Emotional Models . . . 20
2.3.1 Affective Reasoner . . . 20
2.3.2 Em Emotion Architecture . . . 26
2.3.3 Plan-Based Approaches . . . 32
2.3.4 Motivation-Based Behaviour . . . 38
2.4 Emotional Influence on Behaviour . . . 40
2.4.1 Parametric Representations . . . 41
2.4.2 Body Focus, Direction and Cinematography . . . 43
2.4.3 Problem Solving . . . 44
2.4.4 Behaviour Under Stress . . . 46
2.5 Scripting Approaches . . . 47
2.5.1 Improv . . . 48
2.5.2 SCREAM and MPML . . . 49
2.5.3 Social-Psychological Improvisation . . . 49
2.6 System Evaluation . . . 50
2.7 Summary . . . 52

3 Problem Statement **53**
3.1 Overview . . . 53
3.2 Research Objectives . . . 54
3.3 Evaluation . . . 55
3.4 Summary . . . 56

4 System Foundation **57**
4.1 Overview . . . 57
4.2 JAM . . . 57
4.2.1 Procedural Reasoning Architectures . . . 57
4.2.2 JAM Overview . . . 60
4.2.3 Beliefs . . . 60
4.2.4 Goals . . . 61
4.2.5 Plans . . . 62
4.2.6 System Operation . . . 64
4.2.7 Example JAM Trace . . . 67
4.2.8 Modifications to JAM . . . 70
4.3 Summary . . . 72

5 E-JAM Architecture **73**
5.1 Overview . . . 73
5.2 Definitions . . . 76
5.3 Agent Personality . . . 78
5.4 Emotion Dimensions . . . 81
5.4.1 Well-being Emotions . . . 81
5.4.2 Prospect-based Emotions . . . 82
5.4.3 Frustration . . . 83
5.5 Emotional State . . . 84
5.6 Goals and Plans . . . 86
5.7 Emotion Appraisal . . . 88
5.7.1 Prospect Appraisal . . . 88
5.7.2 Event Appraisal . . . 90
5.8 Emotional Impact on Decision-Making . . . 93
5.9 Example E-JAM Trace . . . 95
5.10 Summary . . . 106

6 Abstract Problem Experiments **107**
6.1 Introduction . . . 107
6.2 Experimental Design . . . 108
6.2.1 Plan Libraries . . . 108
6.2.2 Agent Definitions . . . 111
6.2.3 Method . . . 112
6.2.4 Emotion Charts . . . 114
6.2.5 Metrics . . . 115
6.3 Observations . . . 119
6.3.1 Emotional agents behave observably differently than rational agents 120
6.3.2 Emotional agent behaviour is distinguishable from noisy rational behaviour . . . 122
6.3.3 E-Plans cause emotional agents to behave more rationally . . . 124
6.4 Detailed Analysis of Individual Episodes . . . 126
6.4.1 Rationale and Design . . . 126
6.4.2 Emotional State Tendencies . . . 127
6.4.3 E-Plan Selection . . . 127
6.4.4 Utility Computations . . . 128

7 Narrative Experiments **129**
7.1 Introduction . . . 129
7.2 Experimental Design and Procedure . . . 130
7.2.1 The problem-solving scenario . . . 132
7.2.2 Agent Personalities . . . 135
7.2.3 Narrative Generation . . . 136
7.2.4 Questions . . . 140
7.3 Results . . . 141
7.3.1 Experiment 1 . . . 141
7.3.2 Experiment 2 . . . 144
7.3.3 Impact of Emotion Verbs . . . 148
7.4 Discussion . . . 148
7.4.1 E-Plans and emotional influence on utility influence subjects' perception of agents . . . 150
7.4.2 Agent personality and initial emotional state influence subjects' perception of agents . . . 151

8 Summary and Future Directions **153**
8.1 Summary 153
8.1.1 Procedural reasoning with C-Plans and E-Plans 153
8.1.2 Homeostasis 154
8.1.3 Utility computations 156
8.1.4 Winner-takes-all approach 157
8.2 Evaluation 159
8.3 Future Directions 160
8.3.1 Multiple agents 160
8.3.2 Personality improvements 161
8.3.3 Focus of attention 161
8.3.4 Metalevel reasoning 161
8.3.5 Graphical 3D interfaces 162

Bibliography **163**

A Abstract Problem Experiment Data **166**
A.1 Symmetric Scenario 166
A.1.1 Plan Selection Counts 166
A.1.2 Emotion Charts 171
A.1.3 Persistence 204
A.2 Asymmetric Scenario 208
A.2.1 Plan Selection Counts 208
A.2.2 Emotion Charts 213
A.2.3 Persistence 246

B Narrative Experiment Materials **250**

List of Tables

2.1 Examples of Well-being and Fortunes-of-others emotion types [OCC88] . . 16
2.2 Examples of Prospect-based emotion types [OCC88] 17
2.3 Examples of Attribution, Compound, and Attraction emotion types [OCC88] 18
2.4 EEC relations generated for Tom and Dick from Figure 2.6 [Ell92] 24
2.5 High level action PAR [AB02] 41
2.6 Motion factors of the Effort component [AB02] 42

5.1 Summary of emotional impacts on utility variables 80
5.2 Appraisal of E-JAM Prospect-based emotions 82
5.3 World Model representation of intensity 85
5.4 Symbols used in Event Appraisal 90

6.1 Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 116
6.2 Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 118
6.3 Normalized mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. . 119
6.4 Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 121
6.5 Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 121
6.6 Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 123
6.7 Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 124
6.8 Mean persistence for the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience. 125

7.1 The sequence of eight affinity questions used in both of the narrative experiments. 140
7.2 The set of thirteen adjective pairs used in both of the narrative experiments. 141
7.3 Mean affinity question rankings for each group in Experiment 1. Each mean is based on 10 subject responses (1 = "not at all," 7 = "very much"). 142
7.4 Positive adjective selection counts for each group in Experiment 1. Each group is based on 10 subject responses, except where indicated. 143
7.5 Mean affinity question rankings for each group in Experiment 2. Each group is based on 10 subject responses (1 = "not at all," 7 = "very much"). . 145
7.6 Positive adjective selection counts for each group in Experiment 2. Each group is based on 10 subject responses, except where indicated. 145

A.1 Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 166
A.2 Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience. 166
A.3 Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 167
A.4 Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 167
A.5 Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 167
A.6 Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience. 167
A.7 Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience. 168
A.8 Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience. 168
A.9 Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 168
A.10 Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience. 168
A.11 Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 169
A.12 Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 169
A.13 Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 169
A.14 Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience. 169
A.15 Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience. 170
A.16 Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience. 170
A.17 Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 204
A.18 Normalized mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. . . . 204
A.19 Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience. 204
A.20 Normalized mean persistence for the symmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience. 205
A.21 Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 205

A.22 Normalized mean persistence for the symmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience. . 205
A.23 Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 205
A.24 Normalized mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. . 205
A.25 Mean persistence for the symmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 206
A.26 Normalized mean persistence for the symmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. . . . 206
A.27 Mean persistence for the symmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience. 206
A.28 Normalized mean persistence for the symmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience. . . 206
A.29 Mean persistence for the symmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience. 206
A.30 Normalized mean persistence for the symmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience. . . 207
A.31 Mean persistence for the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience. 207
A.32 Normalized mean persistence for the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience. . . . 207
A.33 Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 208
A.34 Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience. 208
A.35 Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 208
A.36 Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 209
A.37 Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 209
A.38 Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience. 209
A.39 Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience. 209
A.40 Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience. 210
A.41 Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 210
A.42 Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience. 210
A.43 Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 210
A.44 Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 211

A.45 Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. . . . 211
A.46 Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience. . . . 211
A.47 Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience. . . . 211
A.48 Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience. . . . 212
A.49 Mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. . . . 246
A.50 Normalized mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. . . . 246
A.51 Mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience. . . . 246
A.52 Normalized mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience. 247
A.53 Mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience. . . . 247
A.54 Normalized mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience. . 247
A.55 Mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. . . . 247
A.56 Normalized mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. . 247
A.57 Mean persistence for the asymmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. . . . 248
A.58 Normalized mean persistence for the asymmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. . . . 248
A.59 Mean persistence for the asymmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience. . . . 248
A.60 Normalized mean persistence for the asymmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience. . . . 248
A.61 Mean persistence for the asymmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience. . . . 248
A.62 Normalized mean persistence for the asymmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience. . . . 249
A.63 Mean persistence for the asymmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience. . . . 249
A.64 Normalized mean persistence for the asymmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience. . . . 249

List of Figures

2.1 Typical synthetic agent architecture 5
2.2 Global structure of emotion types in the OCC model [OCC88] 9
2.3 Global structure of local intensity variables in the OCC model [OCC88] . . 14
2.4 TaxiWorld graphical display [Ell92] 21
2.5 Affective Reasoner processing stages and related representations [Ell92] . . 22
2.6 Example of situation and construal frames [Ell92] 23
2.7 Em emotion architecture [Rei96] 27
2.8 Default Em emotion type hierarchy [Rei96] 29
2.9 The Émile interface with two agents [Gra00] 34
2.10 Interacting with the Mission Rehearsal Exercise [GM04] 36
2.11 Example of a coping elicitation frame [MG02] 38
2.12 Interface to the Simón agent [Vel97] 39
2.13 User interface with choice of thoughts for the agent [MJL00] 44
2.14 User interface of the "dancing mouse" experiment simulation [Bel01] . . . 45

4.1 The PRS architecture [Mye97] 58
4.2 The PRS interpreter loop [Mye97] 58
4.3 The JAM architecture [Hub99] 61
4.4 Example JAM plan showing many of the available actions [Hub99] 63
4.5 Example JAM Intention Structure state [Hub99] 65
4.6 Graphical representation of an example JAM scenario. Ellipses represent goals and boxes represent plans. 66
4.7 Graphical representation of agent behaviour in JAM. Ellipses represent goals and boxes represent plans. Edges from left to right indicate sub-goaling or plan selection. Bold edges and dotted edges from right to left indicate success and failure, respectively. 67

5.1 The E-JAM architecture. Changes from JAM are indicated by italicized elements highlighted in grey. 75
5.2 Emotion dimension data structure for joy/distress 79
5.3 Emotion instance data structure for joy/distress 84
5.4 Example C-Plan to clean the floor 86
5.5 Example E-Plan to dance for joy 87
5.6 Graphical representation of agent behaviour in E-JAM. Ellipses represent goals and boxes represent plans. Edges from left to right indicate sub-goaling or plan selection. Bold edges and dotted edges from right to left indicate success and failure, respectively. Dashed edges indicate where the agent decides to execute an E-Plan instead of a C-Plan. 96

6.1 Graphical representation of "Goal 1" in the symmetric scenario. "Goal 2" and "Goal 3" have identical structures. 109
6.2 Graphical representation of "Goal 1" in the asymmetric scenario. "Goal 2" and "Goal 3" have identical structures. 110
6.3 Emotion chart for the non-noisy emotional agent. The test combination shown is the symmetric sequential scenario with no E-Plans, for a Failure-Failure-Failure problem solving experience. 113

7.1 Graphical representation of the narrative experimental scenario. Grey ellipses and boxes represent goals and plans that will succeed, and white ellipses and boxes represent goals and plans that will fail. 134
7.2 Interaction effects in Experiment 1 for the "relaxed/frustrated" adjective selection 144
7.3 Interaction effects in Experiment 2 for the total adjective selections 146
7.4 Interaction effects in Experiment 2 for the "hopeful/fearful" adjective selection 147

A.1 Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 172
A.2 Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience. 173
A.3 Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 174
A.4 Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 175
A.5 Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 176
A.6 Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience. 177
A.7 Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience. 178
A.8 Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience. 179
A.9 Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 180
A.10 Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience. 181
A.11 Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 182
A.12 Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 183
A.13 Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 184
A.14 Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience. 185
A.15 Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience. 186
A.16 Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience. 187

A.17 Emotion chart for the noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 188
A.18 Emotion chart for the noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience. 189
A.19 Emotion chart for the noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 190
A.20 Emotion chart for the noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 191
A.21 Emotion chart for the noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 192
A.22 Emotion chart for the noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience. 193
A.23 Emotion chart for the noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience. 194
A.24 Emotion chart for the noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience. 195
A.25 Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 196
A.26 Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience. 197
A.27 Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 198
A.28 Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 199
A.29 Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 200
A.30 Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience. 201
A.31 Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience. 202
A.32 Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience. 203
A.33 Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 214
A.34 Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience. 215
A.35 Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 216

A.36 Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 217
A.37 Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 218
A.38 Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience. 219
A.39 Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience. 220
A.40 Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience. 221
A.41 Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 222
A.42 Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience. 223
A.43 Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 224
A.44 Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 225
A.45 Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 226
A.46 Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience. 227
A.47 Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience. 228
A.48 Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience. 229
A.49 Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. 230
A.50 Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience. 231
A.51 Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience. 232
A.52 Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. 233
A.53 Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. 234
A.54 Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience. 235

A.55 Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience. . . . 236
A.56 Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience. . . . 237
A.57 Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience. . . . 238
A.58 Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience. . . . 239
A.59 Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience. . . . 240
A.60 Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience. . . . 241
A.61 Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience. . . . 242
A.62 Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience. . . . 243
A.63 Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience. . . . 244
A.64 Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience. . . . 245

Chapter 1

Introduction

1.1 Motivation

The idea of artificially intelligent agents hearkens back to the first science fiction stories of androids, robotic people who could interact with humans and with each other. Today we have separated the tasks of building physical robots from designing their software, but the basic idea remains the same. A *synthetic agent* is an embodied character who acts independently and interactively in a believable fashion. The believability of such an agent depends on how closely its behaviour and expression match with an observer's expectations of personality and emotions.

Given the goal of "build a believable synthetic agent," how might one go about it? The answer for many researchers has been to design emotional agent systems that attempt to give synthetic agents human-like, emotional behaviour. The approaches taken to build these systems cover a wide range, from complex models of neurophysical interactions to lightweight scripting methods. In this work, we build upon approaches that involve cognitive reasoning and planning aspects, and present a novel framework for building problem solving synthetic agents using a procedural reasoning architecture. Our framework is intended to provide a powerful domain-independent toolkit for researchers and content creators to develop and experiment with synthetic agents.

1.2 Approach

Our synthetic agent framework, named E-JAM, is based on the publicly available JAM procedural reasoning architecture [Hub99]. JAM provides agents with a well-defined process for specifying goals and achieving them using procedural plans. We extend JAM with an emotional model that includes processes and data structures for defining agent personality, generating emotions based on goal and plan outcomes, storing emotional state, influencing the decision-making process, and triggering emotion-based behaviours.

The two key contributions of our framework are the influence of emotions on the decision-making process and the triggering of emotion-based behaviours. The agent's current emotional state influences the expected utility computations of untried plans in the problem solving episode. The agent makes decisions between plans based on utility, and in E-JAM the utility is biased according to the agent's particular emotional state. For example, a character who is frustrated may not care about how much a plan costs, because he just wants to get the task done no matter what. He would lower his consideration of the cost component of utility, so that a plan with high cost would be more likely to be selected when the character is frustrated than when he is contented.

The second important contribution of the E-JAM framework is the inclusion of plan-based behaviour that is triggered by the simulated emotional state, and causes that state to dissipate. Plans that can be executed in response to emotional state compete for execution with plans that can be executed to achieve externally-assigned goals. For example, a frustrated character could decide to stop working on his task and go for a walk to reduce his frustration. However, if his task is very important to him, he may continue working on the task and go for a walk afterwards. This element of the system is inspired by the concept of homeostasis, a property of systems (especially living organisms) to maintain a stable state. When the agent's emotional state becomes non-neutral, emotion-based plans are triggered so that the agent can return to the neutral emotional state.

The E-JAM framework offers a novel approach to synthetic agent modelling by enabling derived emotional state to bias utility computations (affecting performance on the core problem solving task) and to trigger additional behaviours that compete with the execution of the core problem solving task. Both elements can contribute to the development of believable synthetic agents that have problem-solving behaviour as one of their main activities.

1.3 Evaluation

We evaluated the performance of the E-JAM framework with two sets of experiments. First, we used abstract planning problems to confirm the operation of the system and to evaluate the behaviour of E-JAM agents in a simple problem solving domain. On these problems, we found that agents modelled using our key framework elements exhibited clearly different problem solving choices than agents modelled without them.

The second set of experiments consisted of studying the reactions of human subjects to short narrative stories. These stories were generated from the behaviour selections of E-JAM agents in a concrete problem solving scenario. To support these experiments, we added simple natural language generation to the system. We investigated the effects of the key framework elements, as well as the effects of personality and emotional state. We found that agents modelled with the key framework elements are perceived differently than those that are not, on some personality traits.

1.4 Thesis Organization

The remainder of this thesis is organized as follows. Chapter 2 gives a review of related literature. We cover the most influential psychological theories and computational models, a number of important implementations of emotional models, and a variety of work that focuses on particular aspects of synthetic agent systems. Chapter 3 follows with a problem statement that directs the flow of the rest of the thesis.

The procedural reasoning architecture we extended for our work is described in Chapter 4. Chapter 5 follows with a thorough description of the E-JAM framework. We discuss both the theoretical basis for our decisions as well as the details of the implementation.

Chapters 6 and 7 present the evaluation of E-JAM with abstract problem experiments and narrative user study experiments, respectively. In each chapter we discuss the objectives of the experiments, their design, and the results.

We conclude in Chapter 8 with a review of the core elements and contributions of E-JAM. We also identify several areas that were not fully explored and are candidates for future work.

Chapter 2

Literature Review

2.1 Opening Remarks

This research field follows the accepted current AI methodology of *situated agency*. Under this view, a computational agent perceives and acts within some environment. Its decisions for action are based on its perception and a computational model [RN95]. The complexity of the agent's computational model depends on the complexity of the environment and desired agent performance within that environment. Synthetic agent systems are based on a more complex computational model than most other agents, since the environment can be complex and the performance metric ("believability" according to a human observer) can be ill-defined.

Most synthetic agent systems include four components: a method for interpreting stimuli (input); a way to direct agent behaviour and actions, informed by emotional state; a method for expressing emotional state to the world (output); and a computational model of emotions that determines how emotions are generated and managed. Some systems include components such as inter-agent models as part of their computational model. Active research is also carried out that focuses on individual components, since the components can sometimes be made reasonably independent of one another. Figure 2.1 shows the components and organization of a typical synthetic agent architecture. We now discuss the four components of synthetic agent systems.

There are two main sources of input for a synthetic agent. Environmental (external) stimuli and physiological (internal) stimuli are sensed by the agent with its perceptive sensors. The sources of external stimuli are varied – random events in the world, actions of other agents, or aspects of objects. Internal stimuli can include physical drives such as hunger, tiredness, or boredom. The agent interprets (appraises) the stimuli with respect to its current state and personality. The results of the appraisal are used to update the agent's

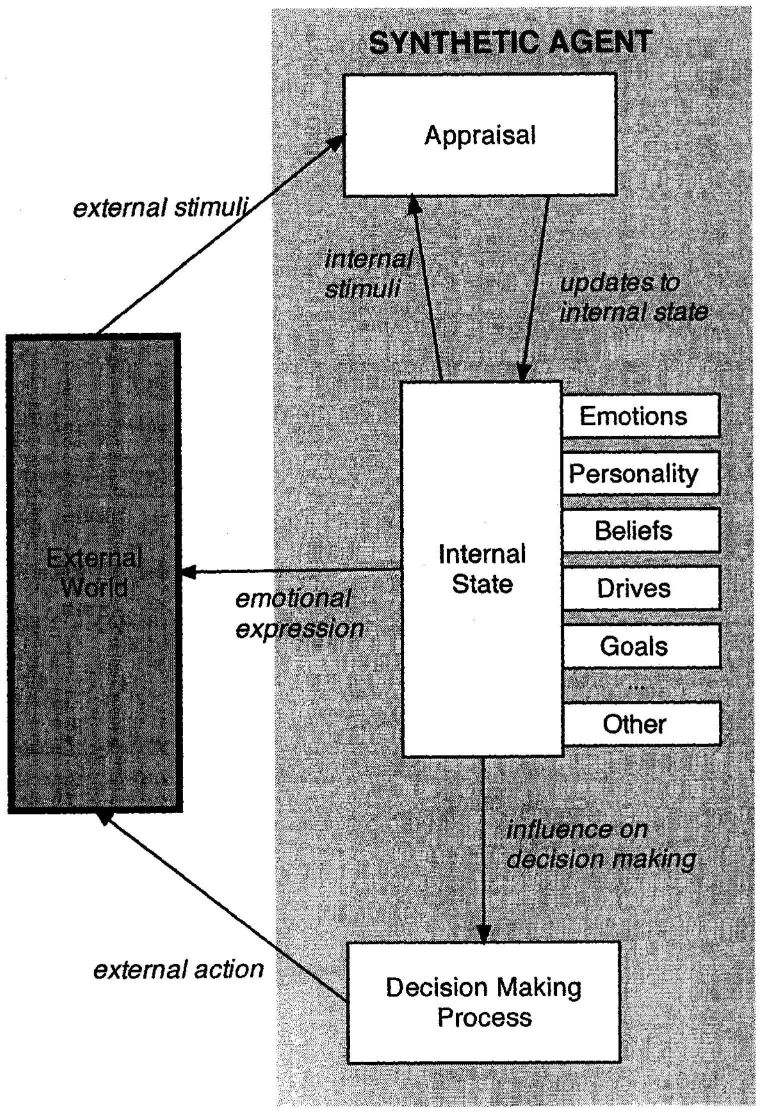

Figure 2.1: Typical synthetic agent architecture

emotional state. How this appraisal is carried out is a major point of synthetic agent research.

A core assumption of synthetic agent research is that agents have an internal emotional state that is updated by the agent's interpretation of stimuli and has some influence on its behaviour. Emotional state is by nature temporary, varying in scope from fleeting emotions to relatively more durable moods. The literature distinguishes emotional state from personality traits, which are long-lasting and relatively consistent. Personality also influences interpretation and behaviour and modulates emotional state. Personality models capture the traits and tendencies that distinguish one person from another. Without difference in personality, there is little difference between the behaviour of two people (or two synthetic agents). Both emotional state and personality are included in the agent's internal state. Typically, the internal state also includes data structures for beliefs about the world and the goals of the agent, along with any other information required by the architecture.

Once an agent's emotional state has been computed, that state can be used to influence the agent's decisions about what actions to take in the world. Since an external observer does not have access to an agent's internal state, it must infer the agent's internal state based on the agent's observable actions. This inferred state is used by the observer to predict or explain further actions on the part of the synthetic agent. However, merely deciding what action to take is usually insufficient for believability. The action must be carried out in the world in a way that convincingly expresses the personality and emotional state of the agent. Many systems finesse this point by including direct emotional expression such as body language facial expressions. Emotional expression, whether direct or through the influence of emotions on choices or actions, is where synthetic agent research is "brought to life" by graphics and animation techniques.

The final component of synthetic agent systems is the computational model of emotions. This component influences almost every aspect of the system because it determines how emotions are generated and how they affect the agent's behaviour. In Figure 2.1, the computational model of emotions can be considered to include all the connections between the other components of the synthetic agent system. The model determines how the appraisal process works, what the internal state includes and how it is organized, how emotions are expressed, and how internal state influences the decision-making process of the agent.

Most research about computational models of emotions for synthetic agents draws from well-established social and cognitive psychological models of emotion and personality. A number of theories from neuroscience also drive the development of computational models of emotions. There are several models to choose from and as many ways to implement them computationally. This leads to many different ways to achieve the target of a believable synthetic agent, but we can roughly divide the approaches into two categories: "shallow" and "deep." An emotional model can range in complexity from simple scripting (shallow) to detailed simulation of the biochemical processes of the human brain (deep). Most approaches fall somewhere in between (but on the shallow side), drawing from cognitive and psychological theory. Several approaches from different categories will be reviewed below.

The rest of this chapter will discuss several important areas in the field of synthetic agent research. We first look at the most influential theories for computational models of emotions. Ortony, Clore, and Collins developed a seminal cognitive model of emotions now called the OCC model [OCC88]. The OCC model, derived from cognitive science research, has had extensive influence on synthetic agent research. We also review a significant personality model called OCEAN, or the Big Five model [JS99]. Next, we discuss several important implementations of emotional models. Two important early systems are the Affective Reasoner by Elliott [Ell92] and the Em emotion architecture by Reilly [Rei96]. Moving into more recent work, we review the planning-based synthetic agent systems introduced by Gratch [Gra00]. General-purpose planners are mature technology in AI, and efforts to incorporate them in synthetic agent systems have been promising. We next consider Cathexis [Vel97], a notable neuroscience-driven synthetic agent system, though the physiological approach to emotional modelling is outside of the scope of our research.

Following our review of major theories and implementations, we explore a variety of work that focuses on different aspects of synthetic agent systems. We examine approaches to emotional influence on behaviour, including parametric representations of action, nonverbal emotional expression, cognitive conflict resolution, and physiological stress. We also discuss probabilistic scripting of emotional behaviour, a shallow approach to synthetic agents. Finally, we consider the problem of effective and accurate evaluation of synthetic agent performance. Most research aims to achieve believability, but there is no generally accepted definition for believability and it is difficult (or perhaps impossible) to measure objectively. We review work that has attempted to determine effective metrics for synthetic agent systems.

2.2 Psychological Theories and Models

In computing science, the most influential models of emotion and personality are based on theories from psychology and cognitive science. These models describe high level concepts such as motivations and intentions, and usually give an organizational structure for emotions and personality. Motivations are distinguished from emotions because there are often non-emotional reasons for people to select a course of action, and because different behaviour can arise from the same emotions. Emotions are generally considered an important influence on motivations, however. Most psychological models define personality and emotion separately for two important reasons. A person's personality is relatively static compared to her emotional state, which can change very quickly. The impact of emotions on behaviour (in the short term) is also significantly greater than the influence of personality. Some theories define the concept of "mood" to bridge the gap between personality and emotion. A person's mood is thought to change more quickly than her personality, but not as rapidly as her emotions.

2.2.1 OCC Model

The OCC model is characterized by its authors as "a cognitive theory concerning the origin of the emotions" [OCC88]. To achieve this lofty specification, the model consists of four major related components: a global structure specifying the relationship between different emotions, an appraisal process for generating emotions, a set of factors that influence emotion intensity, and a detailed analysis of individual emotion types and categories. The OCC model is an influential theory that has been realized in some implemented systems. We discuss one such system, developed in part by the authors of the OCC model, in Section 2.3.1.

Structure of Emotions

In the OCC model, the notion of an *emotion type* is used to distinguish between distinct kinds of emotions. Emotion types can have several emotion words associated with them. For example, the emotion type *joy* can be referred to in English by several words such as "happiness," "contentment," "ecstasy," and so on. Each of these words suggest a different intensity to the emotion type or place a different emphasis on the nature of the emotion, but they all refer to the same emotion type.

Defining a limited set of emotion types leads immediately to the problem of determin-

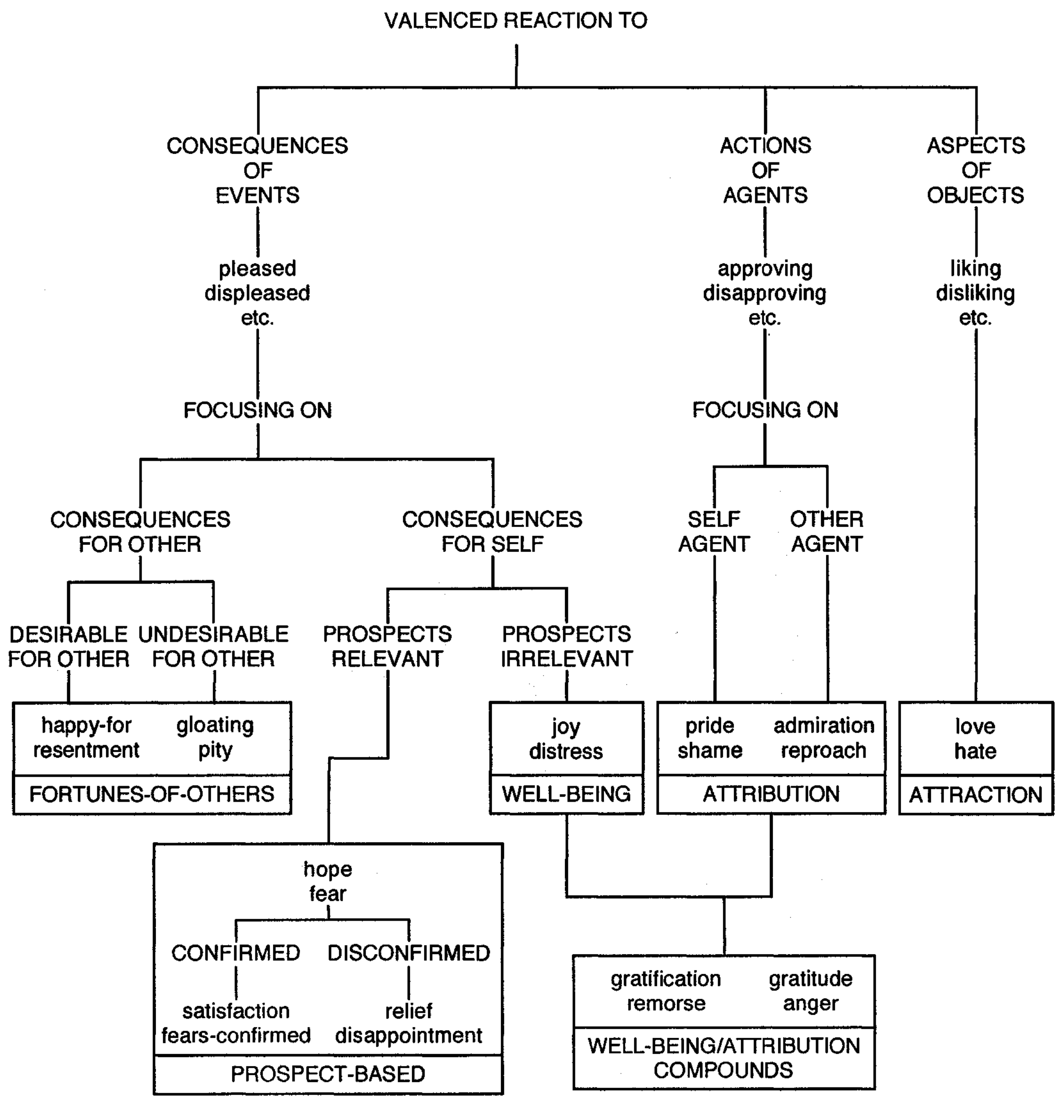

Figure 2.2: Global structure of emotion types in the OCC model [OCC88]

ing what emotion types exist. The OCC model gives a logical hierarchical structure to emotion types that differentiates between representative groups or clusters of emotion types by simple *eliciting conditions*. An eliciting condition is a situational description of the requirements for an emotion to be triggered. For example, if a friend wins the lottery, that situation would match the eliciting condition "an event presumed to be desirable for someone else." The logical structure is presented as a tree with emotion types denoting valenced reactions (see Figure 2.2). The main assumption that drives the structure of the OCC model is that agents can focus on three major aspects of the world: *events*, *agents*, and *objects*. All emotions are valenced reactions to one of these major aspects. Accordingly, the three main branches of the global structure are driven by the three major aspects: consequences of events, actions of agents, or aspects of objects. Associated with each main branch is a general class of emotional reaction. An agent may be pleased or displeased about an event, approve or disapprove of an agent's actions, or like or dislike an object. By proceeding down the tree we can see how the emotions are organized.

Several broad groups of emotion types (such as the Well-being emotions) are denoted in the structure with boxes. Each of the emotion types in a group are structurally related in terms of eliciting conditions or situational prerequisites, and are distinguished from one another by different variable settings for the conditions. For example, "joy" requires a desirable event while "distress" requires an undesirable event. Both emotion types have the eliciting condition that an event must occur, but they differ in the desirability variable setting for the condition. Another characteristic of emotion groups is that the emotions in a group represent a cluster of closely related emotions. While the emotions are structurally related, they are distinguished from one another by their intensity or by how they are manifested.

The global structure diagram in Figure 2.2 illustrates the eliciting conditions for the emotion groups defined in the OCC model. By tracing downwards we can see the emotions that require certain sets of conditions, and by tracing upwards we can determine what eliciting conditions apply to a given emotion type. The diagram does not give the details of the values of each condition, however. We discuss each emotion group in detail later in this section.

Appraisal Process

The intensity of an emotion in the OCC model is determined by an appraisal process that is structured along the lines of the three main aspects of the world. Agents are assumed to have a knowledge representation system called the *appraisal structure* which consists of complex goals, standards, and attitudes (sometimes called preferences). Associated with these conceptual structures are three central variables that influence emotion intensity.

Goals represented in the appraisal structure can range from high level aspirations or general concerns (such as "have happy family life") to low level immediate goals (for instance, "open the door"). Goals have incoming and outgoing connections to other goals. A lower level goal has outgoing connections that go into higher level goal(s) to represent that the low level goal is a part of or requirement for the success of the high level goal(s). The directed nature of the connections leads to a roughly tree-like shape for the goal structure, with few high level goals and many low level goals. This structure is similar to most goal/sub-goal structures in cognitive research. Goals have value corresponding to their importance to the agent and how they contribute to the achievement of higher level goals. The OCC model makes a distinction between three kinds of goals: *Active-pursuit* goals (**A**-goals), *Interest* goals (**I**-goals), and *Replenishment* goals (**R**-goals). **A**-goals represent states that an agent wants to achieve or obtain, **I**-goals represent things that an agent wants to see happen (even if the agent cannot exert a direct influence), and **R**-goals represent requirements that have a cyclical nature and cannot be discarded when they are realized. **R**-goals correspond to maintenance goals in AI systems. A further distinction between goals is that of *all-or-none* goals or *partially attainable* goals. Some goals can either be achieved or not (all-or-none), while others can be achieved to some degree. This distinction comes into play when considering emotions arising from complete or partial success of a goal.

Aside from goals, the appraisal structure includes representations of standards and attitudes. Standards represent morals, conventions, and values that determine how one believes people ought to behave. They are frequently justified in terms of social considerations and so have a different, more intrinsic source of value than goals, whose value is determined from a more personal perspective. Attitudes are defined as the dispositions an agent has toward liking or disliking certain objects. They are defined without reference to goals or standards, but can nevertheless have a significant contextual influence on emotional responses related to goals and standards.

The three central variables of the OCC appraisal process are associated with the three aspects of valenced reactions and with the three types of objects in the appraisal structure. *Desirability* is associated with reactions to events, which happen in relation to goals. *Praiseworthiness* is associated with reactions to the actions of agents and is influenced by standards. *Appealingness* is associated with reactions to objects, which are connected to attitudes. Each central variable affects the intensity of all emotions that involve its associated aspect. For example, all else being equal, an agent would be more joyful about the occurrence of a highly desirable event than of a slightly desirable event.

The central variable of desirability has two aspects: the degree to which the related event has beneficial consequences, and the degree to which it has harmful consequences. Events are usually related to the goals of an agent. Every goal in an agent's appraisal structure has a value that is determined primarily by its position in the structure and how it helps or hinders the achievement of other goals. Thus, the greater the value of the goal, and the greater the value of the higher-level goals that the goal facilitates, the greater the desirability associated with achieving that goal. Likewise, undesirability is closely related to the value of the higher-level goals that the goal hinders. Since a goal can facilitate some higher-level goals (to varying degrees) and hinder other higher-level goals (to varying degrees), the event of achieving the goal can have both desirable and undesirable aspects.

The actions of other agents, evaluated with respect to standards, give rise to the central variable of praiseworthiness. When other agents do things that uphold our own valued standards, we consider their actions praiseworthy. On the other hand, when their actions violate our standards, they are considered blameworthy. The degree of praiseworthiness or blameworthiness is determined by both the degree to which standards are upheld or violated and by the value of the associated standards. It is important to note that while agents are usually people and considered responsible for their actions, several exceptions can be (and are often) made. For example, children are not typically blamed for normally blameworthy actions because they do not completely understand the consequences of their actions. Standards acquire their value both from (generalized) standards higher in the hierarchy, as well as from connections to high-level I-goals that try to maintain standards-related world states. While the standards in the OCC model are powerful, they require a very complex and sophisticated world and implementation to fully realize.

The third central variable is appealingness, which relates to how people evaluate objects with respect to their attitudes. The values of attitudes represent the degree of disposition towards liking or disliking a certain sort of object (or conceptual representation of the object). The value of the appealingness variable is directly related to the value of the associated attitudes. Appealingness is especially relevant to emotions with low cognitive content because it is not affected by the significance of goal-relevant consequences.

Other Variables Influencing Intensity

Each central variable is uniquely associated with one aspect of an agent's reactions to the world: events, agents, or objects. There are several other factors included in the OCC model which influence emotional appraisal. These factors are divided into two main groups, global variables and local variables. Global variables influence the intensity of all emotion types, while local variables (which, technically, include the central variables) affect only certain emotion groups or emotion types. A requirement for intensity variables is that they influence intensity independently, not just affect the value of a different variable. The four global variables considered in the OCC model are *sense of reality*, *proximity*, *unexpectedness*, and *arousal*.

The sense of reality variable is linked to the degree that an emotion-inducing situation seems real to an agent. The more real an agent believes a situation, the more intense the associated emotions. Proximity is related to the sense of reality variable, but is not the same. It is determined by how "close" an emotion-inducing situation is to the agent. Closeness can be considered in terms of temporal, spatial, or psychological proximity. The closer a situation is to an agent, the more intense the associated emotions. The global variable of unexpectedness is well-recognized in emotional research. The more surprise associated with an event, the greater the intensity of the emotion. Unexpectedness is distinguished from the idea of likelihood because an event can be completely unexpected either if an agent believes it is unlikely, or if the agent has not even considered the event at all. The last global variable is physiological arousal, and is cognitive in nature. However, it has important positively correlated effects on the intensity of emotions. Arousal can be increased by non-emotional causes and decays slowly, so it has an influence on emotional reactions removed in time from the original cause of arousal.

The OCC model defines a number of local variables that influence only specific emotion groups. The three central variables of the model, desirability, praiseworthiness, and appealingness, are local variables that influence the Event-based emotions, Attribution emotions, and Attraction emotions, respectively. Figure 2.3 shows the influence of all the local intensity variables on the global emotion structure. It represents an inheritance property of variables, wherein each variable affects all emotions in the structure linked below its place in the structure. The unique configuration of local variables is part of the distinctiveness of the emotion types placed lower in the structure.

As indicated above, all the Event-based emotions are influenced by the desirability

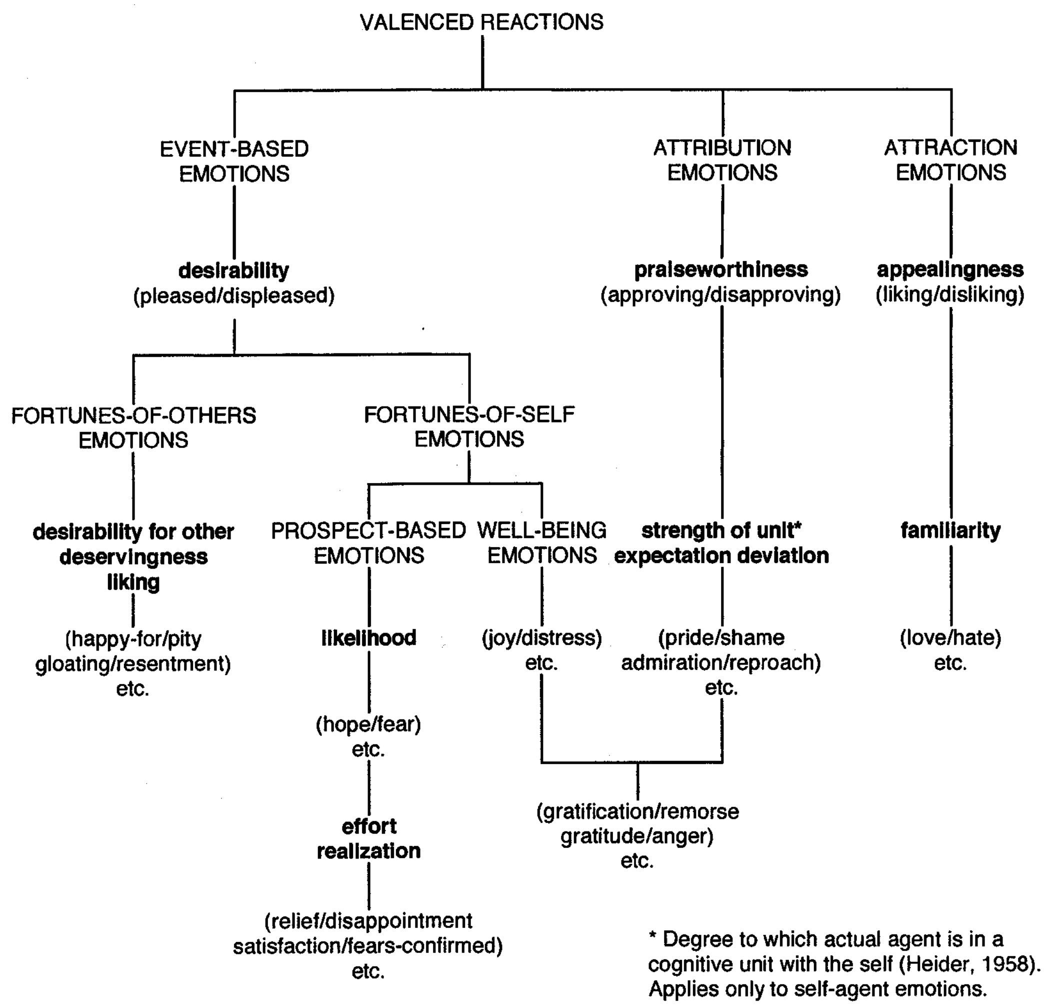

Figure 2.3: Global structure of local intensity variables in the OCC model [OCC88]

variable. The relatively simple Well-being emotions are not influenced by any other local variables, but the Prospect-based emotions and Fortunes-of-others emotions each involve three additional local variables. All Prospect-based emotions are influenced by *likelihood*, which is related to the agent's belief about the probability that an event will occur. The higher the likelihood, the more intense the emotion that is experienced in relation to the prospected event. The Prospect-based emotions that involve reactions to the confirmation or disconfirmation of prospects are influenced by *effort* and *realization*. Effort includes the ideas of physical or mental exertion as well as more materialistic kinds of costs. Realization captures the degree to which an event (confirmed or disconfirmed) is actualized.

The three additional local variables that influence the Fortunes-of-others emotions are *desirability-for-other*, *liking*, and *deserving*. The Fortunes-of-others emotions are activated by reactions to events that have an effect on another person. These events and their consequences can be seen as desirable or undesirable for the other person. Another contribution to the intensity of these emotions is the degree to which the agent likes the other person. The *liking* variable here is considered in the momentary sense of liking, as opposed to a disposition towards liking. The last variable associated with the Fortunes-of-others emotions is the degree to which the agent believes the other person deserves the event affecting him or her.

The Attribution emotions are affected by the central *praiseworthiness* variable, along with two other local variables. The variable of *strength of cognitive unit* is included to represent the effect of an agent's association with some other person or group. A cognitive unit is a context-sensitive bond between a person and a larger group (such as one's country) or another person (such as a fellow student). The second variable influencing the Attribution emotions is that of *expectation-deviation*, a special form of unexpectedness. Expectation-deviation is specifically related to the degree which people deviate from their expected behavioural norms, whether the expectations are in terms of a particular person, a person of their type, or a person in their role.

Aside from the central variable of *appealingness*, the Attraction emotions have just one additional local variable. *Familiarity* reflects the number of exposures that an agent has to a particular object. According to the OCC model, all other influences on the intensity of liking can be explained in terms of appealingness or through global variables.

Emotion Type	Example from OCC Model
joy	"The man was pleased when he realized he was to get a small inheritance from an unknown distant relative."
distress	"The driver was upset about running out of gas on the freeway."
happy-for	"Fred was happy for his friend Mary because she won a thousand dollars."
sorry-for	"Fred was sorry for his friend Mary because her husband was killed in a car crash."
resentment	"The executive resented the large pay raise awarded to a colleague whom he considered incompetent."
gloating	"Political opponents of Richard Nixon gloated over his ignominious departure from office."

Table 2.1: Examples of Well-being and Fortunes-of-others emotion types [OCC88]

Emotion Types

The OCC model includes twenty-two emotion types organized into six emotion groups. Most emotion types fall under the event-based part of the structure, with the rest concerning reactions to agents or objects. The model gives a specification for emotion types that includes five major components. The first component is the emotion *type identification*, which is used as simple label for the emotion type (e.g. joy). The second component is the *type specification*, which summarizes the necessary conditions for the activation of the emotion type. Next is a selected list of emotion words (*tokens*) that are related to the emotion type. The fourth component is a list of the local *variables affecting intensity* of the emotion. An *example* is given as the final component of an emotion specification. For brevity, we will not fully describe each emotion type here, but we highlight the most important points of each one. The OCC model includes a detailed description of each emotion type and the problems associated with their appraisal. Examples of each emotion type are given in Tables 2.1, 2.2, and 2.3.

The Well-being emotions have a fairly straightforward structure. There are two emotions in this group, with the representative labels "joy" and "distress." Joy is experienced when the agent is pleased about a desirable event, while distress occurs when the agent is displeased about an undesirable event. Joy and distress emotions arise when an agent focuses only on the desirability or undesirability of an event, not the other aspects of the event. Since the Well-being emotions occur in response to actual events, as opposed to the prospect of events, they are considered "pure" states of being pleased or displeased. Table 2.1 gives examples of the joy and distress emotion types.

The Fortunes-of-others emotions are concerned with reactions to events that primarily affect other people. An event can be desirable or undesirable for another agent, and the

Emotion Type	Example from OCC Model
hope	"As she thought about the possibility of being asked to the dance, the girl was filled with hope."
fear	"The employee, suspecting he was no longer needed, feared that he would be fired."
satisfaction	"When she realized that she was indeed being asked to go to the dance by the boy of her dreams, the girl was gratified."
fears-confirmed	"The employee's fears were confirmed when he learned that he was indeed going to be fired."
relief	"The employee was relieved to learn that he was not going to be fired."
disappointment	"The girl was disappointed when she realized that she would not be asked to the dance after all."

Table 2.2: Examples of Prospect-based emotion types [OCC88]

agent who is assessing the situation is generally either pleased about it or displeased about it. When these characteristics are congruent with one another, Good-will emotions result, but when they are not congruent, Ill-will emotions tend to arise. The Good-will emotions include "happy-for" (pleased about an event desirable for another) and "sorry-for" (displeased about an event undesirable for another). The two Ill-will emotions are "resentment" (displeased about an event desirable for another) and "gloating" (pleased about an event undesirable for another). Table 2.1 gives examples of the Fortunes-of-others emotion types.

When events are expected by an agent, the Prospect-based emotions come into play. The base of this emotion group is the pair of emotion types labelled "hope" and "fear." These emotion types may be active when events are unconfirmed or yet to occur. Hope is experienced when an agent is pleased about the prospect of a desirable event, and fear occurs when an agent is displeased about the prospect of an undesirable event. Once the occurrence of a prospected event is confirmed or disconfirmed, two further sets of two emotion types each are applicable. The Confirmation emotions include "satisfaction" and "fears-confirmed," while the Disconfirmation emotions include "relief" and "disappointment." Satisfaction is experienced when an agent is pleased about the confirmation of the prospect of a desirable event, and fears-confirmed is activated when an agent is displeased about the confirmation of the prospect of an undesirable event. Correspondingly, relief arises when an agent is pleased about the disconfirmation of an undesirable event, while disappointment is experienced when an agent is displeased about the disconfirmation of a desirable event. These emotions are influenced by the intensity of the related hope or fear emotion, as well as other variables. The effect of variables that influence the hope/fear emotion are passed through as part of its intensity to the confirmation and disconfirmation emotions. If there is no associ-

Emotion Type	Example from OCC Model
pride	"The woman was proud of saving the life of a drowning child."
self-reproach	"The spy was ashamed of having betrayed his country."
appreciation	"The physicist's colleagues admired him for his Nobel prize-winning work."
reproach	"Many people despised the spy for having betrayed his country."
gratitude	"The woman was grateful to the stranger for saving the life of her child."
anger	"The woman was angry with her husband for forgetting to buy the groceries."
gratification	"The man was gratified by his daughter's achievements."
remorse	"The spy felt remorse at the damage he had done in betraying his country."
liking	"Mary was filled with affection as she gazed at her newborn infant."
disliking	"John disliked the concert so much that he left in the middle."

Table 2.3: Examples of Attribution, Compound, and Attraction emotion types [OCC88]

ated hope/fear emotion, the confirmation and disconfirmation emotions will not be affected. Table 2.2 gives examples of the Prospect-based emotion types.

The Attribution-of-responsibility emotions result from reactions to the actions of agents. The central variable of praiseworthiness affects the Attribution emotions by introducing the idea of approval or disapproval of an agent's actions. The Attribution emotions are divided into two subgroups, those resulting from actions of the self, and those resulting from actions of other agents. "Pride" arises when an agent approves of its own praiseworthy action, while "self-reproach" (or shame) results when an agent disapproves of its own blameworthy action. The Attribution emotions dealing with other agents are "appreciation," approving of someone else's praiseworthy action, and "reproach," disapproving of someone else's blameworthy action. Table 2.3 gives examples of the Attribution emotion types.

Certain distinguishable emotions result when agents focus on *both* the consequences of an event and the actions of an agent responsible for the event. These compound emotions have eliciting conditions that are the conjunction of the eliciting conditions for an Attribution emotion and an Event-based emotion of the same valence. The self-oriented compound emotions are "gratification" (pride + joy) and "remorse" (shame + distress), and the other-oriented ones are "gratitude" (admiration + joy) and "anger" (reproach + distress). The compound emotions tend to be more intense than their constituent emotions because they are influenced by intensity variables from two sources. Table 2.3 gives examples of the compound emotion types.

The final major group of emotions defined in the OCC model is the Attraction emotions. They are momentary reactions of disliking and liking objects or aspects of objects. The Attractions emotions include "liking" an appealing object and "disliking" an unappealing object. While structurally simple, these emotions are quite complex when considered in context of appealingness, attitudes, and tastes. Appealingness is strongly influenced by dispositional likes and dislikes, which often derive from categorizations of objects and characteristics of individual objects. Table 2.3 gives examples of the liking and disliking emotion types.

Influences and Evaluation

The OCC model has made a strong contribution to the field of synthetic agent research by approaching the problem from a cognitive science background. One goal of the OCC model was to provide a strong foundation for a model of emotions that was computationally tractable. The computational model of emotions is well-defined and elements of it have since been used and extended by a great number of synthetic agent systems. The appraisal process and emotion types of the model are not ad-hoc, but drawn from cognitive research. This scientific approach has led researchers in this field to work more rigourously. For our own research, we draw on several elements of the OCC model, including emotion types and intensity variables.

The authors of the OCC model did not provide an implemented system with the model, but further work has implemented the model to varying extents. The theory has been tested in part with the implementation of the Affective Reasoner, discussed in Section 2.3.1. Nearly all other synthetic agent systems since the publishing of the OCC model have either been influenced by it or have implemented parts of it, and their results reflect upon the model as well.

2.2.2 OCEAN Personality Model

The OCEAN personality model abstracts the differences between individual personalities into five basic personality traits: openness, conscientiousness, extraversion, agreeableness, and neuroticism [JS99]. These five variables were distilled from lists of thousands of traits in order to make factor analysis practical. The OCEAN model is known as the "Big Five" model because each of the factors is broadly defined to include a large number of specific characteristics. John and Srivastava [1999] provide a succinct definition for each factor:

> Briefly, Extraversion implies an energetic approach toward the social and material world and includes traits such as sociability, activity, assertiveness, and positive emotionality. Agreeableness contrasts a pro-social and communal orientation towards others with antagonism and includes traits such as altruism, tender-mindedness, trust, and modesty. Conscientiousness describes socially prescribed impulse control that facilitates task- and goal-directed behaviour,

such as thinking before acting, delaying gratification, following norms and rules, and planning, organizing, and prioritizing tasks. Neuroticism contrasts emotional stability and even-temperedness with negative emotionality, such as feeling anxious, nervous, sad, and tense. Finally, Openness to Experience (vs. closed-mindedness) describes the breadth, depth, originality, and complexity of an individuals mental and experiential life [JS99].

The OCEAN personality model is used as part of several emotional computing systems, including the EMOTE system described later in this chapter [AB02].

2.3 Implementations of Emotional Models

This section details several important and influential systems that implement emotional models. Two major implementations are the Affective Reasoner and Em. Both systems have some basis in the OCC model, but they approach the problem of synthetic agents from different directions. Two more implementation approaches are examined, a high level domain independent plan-based architecture and a low level motivation-based system that incorporates physiological theories.

2.3.1 Affective Reasoner

The Affective Reasoner by Clark Elliott is the first significant implementation of the OCC emotional model [Ell92]. Elliott explores emotional reasoning by simulating multiple agents acting in a rich environment. The agents in the Affective Reasoner have the ability to interpret situations in the environment and generate emotions, model the emotions of other agents, and express emotions as actions in the environment. The three primary components of the Affective Reasoner are the domain-independent reasoning component, a domain-specific world simulation, and a domain-specific graphical interface. The first domain developed for the Affective Reasoner was *TaxiWorld*, a simulation of taxi drivers, passengers, and other agents operating on a graph representation of Chicago. The agents move about the environment and interact with one another. For example, a taxi driver can go to Northwestern University, pick up a passenger, and take her to the airport. TaxiWorld is a fairly rich environment, including forty-five types of events that can happen in the world. For example, the taxi driver could get stopped by a police officer and receive a speeding ticket. The system also defines approximately forty different personality specifications for agents. For instance, one taxi driver might be angry and rude to passengers who leave a small tip,

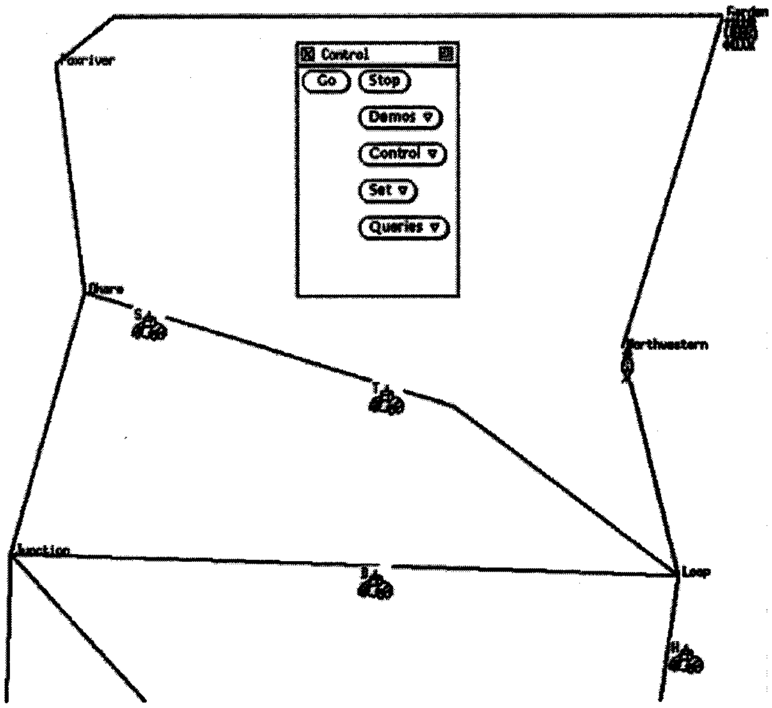

Figure 2.4: TaxiWorld graphical display [Ell92]

while another might not care. Twenty-four emotion types (the twenty-two defined in the OCC model, plus attraction/attribution compounds for "love" and "hate") and sixty action responses are available to the agents.

For the TaxiWorld domain, there are two types of input into the system. The first type is the static configuration of the world. The object domain can be extended by adding simulation events and event handlers, emotional interpretations of situations, emotion expressions, and personality types. Simulation sets (start states) are also defined at this stage. The second type of input is dynamic manipulation of the system through a graphical user interface. The interface shows a graphical representation of the world, with agents represented as icons moving around a graph. Figure 2.4 shows a screen shot of the interface, with taxis moving around and passengers waiting at locations. The user can control parameters of the simulation through a menu, set global moods for the agents, and observe the experiences and interactions of the agents in real time. The running system can also be interacted with directly to control the simulation at the finest level of detail desired. While the main output of the system is graphical, all of the simulation events and emotion histories can be saved for further analysis.

The Affective Reasoner goes through a process of interpretation, emotion generation, action, and observation on every potentially emotion eliciting event in the simulation. Fig-

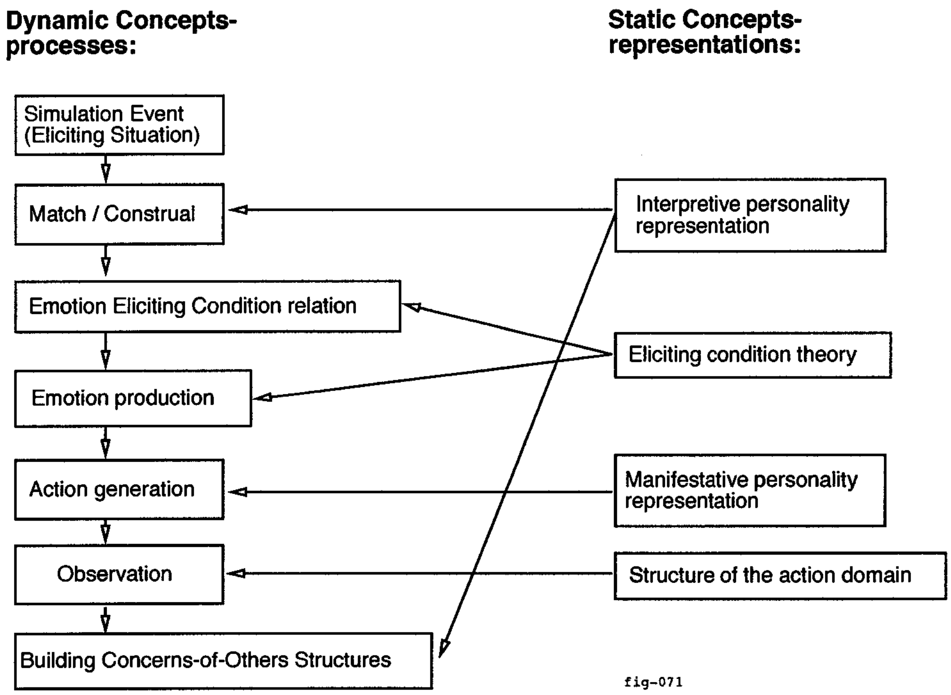

Figure 2.5: Affective Reasoner processing stages and related representations [Ell92]

ure 2.5 shows an overview of the processing stages and associated data structures of the system. Most of the data structures in the system are based on *frames*, a structure that represents a stereotypical situation [Min74]. Events in the simulation are defined by emotion eliciting situation frames. These frames specify the type of event and domain specific information about the event. Agents interpret a situation with respect to their individual *interpretive personality* and concerns. The interpretive personality consists mainly of a hierarchy of *construal frames* which are used to interpret emotion eliciting situations. Construal frames include an event type and domain specific information, similar to event frames. However, construal frames also contain *predicate* and *blocked* slots which determine how an event is interpreted. Predicate slots specify a function that decides whether a given event frame will concern an agent, while blocked slots denote whether or not a goal will be blocked by a matching event frame. Agents also have a Goals, Standards, and Preferences (GSP) hierarchy of frames which can contribute inherited information to construal interpretations.

Figure 2.6 shows an example of how three different agents interpret a situation. The situation frame is shown in the box on the right and consists of a last-second touchdown in a football game, giving Northwestern a 28-27 win over Illinois. The three agents, Tom, Dick, and Harry, each have construal frames that match the event described in the situation frame. Tom is a Northwestern fan and has a goal that Northwestern should win with a

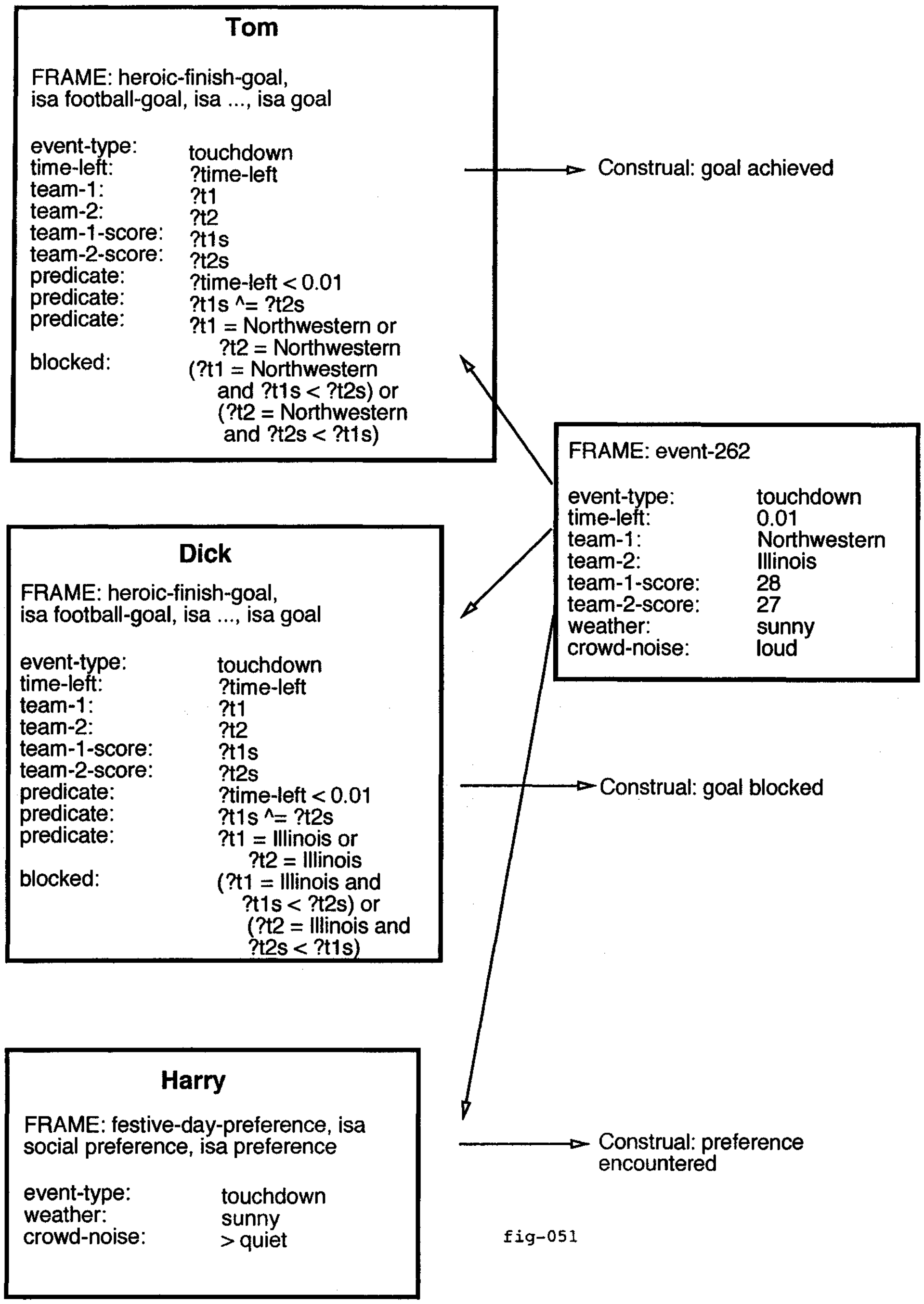

Figure 2.6: Example of situation and construal frames [Ell92]

self	Tom	Dick
other	none	none
desire-self	desire	undesire
desire-other	none	none
pleased	none	none
status	none	none
appraisal	none	none
appeal	none	none
responsible agent	none	none

Table 2.4: EEC relations generated for Tom and Dick from Figure 2.6 [Ell92]

last-second touchdown. Dick has a similar goal, except that he is an Illinois fan and wants them to win with a last-second touchdown. Harry, on the other hand, does not care about the teams in particular, but is interested in the weather being sunny and the crowd being excited. We can see that in this example, Tom construes the situation as achieving his goal, Dick sees his goal blocked, and Harry's preference is satisfied.

After applying the interpretive personality construal frames to the emotion eliciting situation frames, a nine-attribute *Emotion Eliciting Condition (EEC)* relation is generated. EEC relations are made up of variables from the OCC model such as agent identity, desirability, appealingness, etc. Specific emotion types have eliciting conditions that are matched against this relation using domain-dependent rules defined by the researcher and domain-independent rules drawn from the OCC model. If a valid match is made, it leads to the manifestation of an emotional state in the agent. The Affective Reasoner side-steps the issue of having simultaneous conflicting emotions by removing action responses that are incompatible with one another. Table 2.4 shows the EEC relations generated for Tom and Dick in the previous example. Tom's EEC relation shows "desire-self" to be desirable because his goal is achieved, while Dick's EEC relation shows it to be undesirable because his goal is blocked. These EEC relations may lead to joy and distress emotions for the agents, respectively.

The manifestive personality of an agent directly influences the response actions that the agent takes given a situation and emotional state. Response actions are organized along about twenty categories such as expressive, somatic, communicative (verbal), etc. The actions associated with a category may be simple expressions, templates that use bound variables (such as glaring at some agent), or more complex "mini-plans." An agent's manifestive personality is defined by enabling or disabling action response categories (or temperament traits) for positive or negative valenced emotions. For example, an agent described as friendly and shy might have the trait "behavioural (towards-animate)" enabled for positive emotions, and the trait "communicative (verbal)" disabled for positive emotions. The result would be that the agent would choose action responses like smiling and laughing, but not action responses like cheering and bragging. The response categories available to the agent will be limited by the temperament traits that are enabled. The agent's emotional state, manifestive personality, and bound world variables are used to select a response action. Conflicts between incompatible actions are resolved, and remaining actions are instantiated and executed in the world according to their domain-specific definition.

The final processing step of the Affective Reasoner involves observing the actions of other agents and building representations of their concerns. The system uses a knowledge acquisition program to give agents the ability to reason about the actions they observe other agents taking in the world. The agents can observe situations and response actions of other agents and make an assessment of which emotions the agents are feeling. As examples are observed in the world, they are saved and updated to build a representation of the typical actions taken and emotions felt by other agents. The *Concerns of Other* (COO) databases capture knowledge about how other agents interpret the world, so that an agent can try to predict how they will respond to a given action or situation. COO databases are represented using the same data structures and models as an agent's own GSP and personality representations. At first, agents believe that all other agents think the same way they do (providing defaults). When another agent behaves in ways that differ from how the first agent would behave, the first agent updates his COO database for the other agent (and agents who seem similar) to capture a better representation of the concerns of the other agent. For example, Frank might observe that Joe is always unhappy when the Oilers lose. Frank might infer that Joe construes the situation as blocking one of his goals. Frank can do further reasoning about Joe's motives – is he an Oilers fan, or is he just betting money on them to win? As Frank observes further behaviour from Joe (or asks him about it), he can update his COO database for Joe to improve his beliefs about Joe's goals and emotions. The Affective Reasoner even represents COO databases to multiple degrees; i.e. it represents the beliefs about another agent's beliefs about yet another agent. These multi-level COO databases are called *satellite COOs* and are updated with the same process as single-level COOs. For example, suppose Frank observes Joe becoming angry when Bob asks him a question about an Oilers' loss. If Frank thinks that Joe is betting on the Oilers to win, he might reason that Joe thinks that Bob is trying to get him in trouble for illegal gambling.

The Affective Reasoner is an important contribution to the field because it is an effective platform for synthetic agents who reason about situations in the world to generate emotions. It also explores the problems associated with multiple interacting agents by giving them the ability to reason about the emotions and actions of one another in an emotional context. It is the first major implementation of the OCC model and thus more precisely specifies details about issues that may have been vague in theory. However, the Affective Reasoner was not subject to a rigourous experimental evaluation. It was evaluated on an anecdotal basis, and the main contribution was the implementation itself, not any results generated from the system. The system is limited by its very complexity; it was not designed to be used by artists, but by computing science researchers. Another limitation is that the architecture strictly follows the OCC model and thus does not allow for non-cognitive generation of emotions. It is possible that for these reasons, the Affective Reasoner did not see wider use or evaluation.

2.3.2 Em Emotion Architecture

While the Affective Reasoner focuses on a cognitive reasoning approach to generate and use emotions, Reilly's Em emotion architecture [Rei96] takes an artist-oriented approach. His work aims to make it easy for artists to create synthetic agents who are not necessarily intelligent, rational, or realistic, but who have strong personalities that make them interesting as characters. Also explored in Em is the social interaction between agents and how relationships affect emotions and vice versa. Reilly argues that by studying both emotions and social behaviours, the system offered to designers can become more rich and interesting. However, the social behaviour aspect of this work will not be treated fully here. The main application envisioned for the Em system is interactive drama, which comes in many forms. Generally, interactive drama involves human participants, a world and story to explore and influence, and rich virtual characters for the user to interact with. The purpose of the Em emotion architecture is to give synthetic agents more believable behaviour through the use of emotional and social modelling.

Three different simulation systems were built to test the Em emotion architecture and social behaviour methods. All of the systems involve text-based user interaction with multiple agents in a simple physical world. The Em architecture can be used with a graphical interface, but text-based interaction was used to focus on the exploration of more complex behaviours than could be easily represented graphically. The textual input and output of the systems is fairly limited, using keywords and templates, but are adequate for the purposes of the research. In "Robbery World," the user takes the role of a police officer who tries

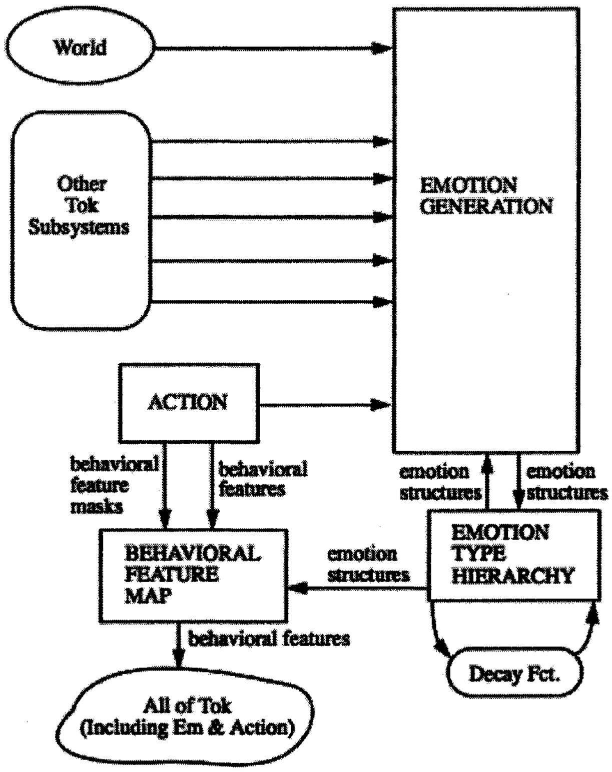

Figure 2.7: Em emotion architecture [Rei96]

to stop a gunman from holding up a convenience store cashier. This domain is meant to test the ability of the Em architecture to model complex emotions. The other two domains, "Office Politics" and "The Playground," are intended to test social interaction between synthetic agents. In each scenario the user must interact with characters using a variety of social behaviours.

Reilly defines an *emotion architecture*, such as Em, as a "general framework for creating emotion systems for particular agents" [Rei96]. Em was targeted as part of the Tok agent architecture used by the Oz Project at CMU [BLR92a, BLR92b]. An *emotion system* is a specific instantiation of an emotion architecture and can correspond roughly to the personality of a particular agent. Essentially, an emotion system fills in the architecture to produce reasonable behaviours. Input from the world can lead to specific emotional episodes that cause an agent's emotional state. *Emotion generators* determine how emotional state (represented by emotion structures) is produced from world input. *Emotion structures* are organized in a hierarchy, can be combined with one another, and decay over time. They also map to behavioural features which lead to agent actions and output to the world. The Em framework is very general and provides an open structure for creating synthetic agents.

However, its flexibility can be a problem in itself, because it is difficult to determine how to begin to construct a character in such a framework. To solve this problem, Reilly provides a default emotion system with generators based on the OCC model. In addition, his dissertation on Em serves as a guide for artists to create specialized emotion systems. Figure 2.7 shows an overview of the Em architecture in the context of Tok.

In the default emotion system provided with Em, emotion generators receive input in the form of sensory data, sense memory, goals, standards and attitudes, body state, and goal processing information. Other available input that is not directly used in the default system includes social relationships, models of other agents, and emotion structures. Emotion generators are written in a scripting language and can be as complex as desired. The standard library of emotion generators is based on the OCC cognitive emotion model. It includes all the emotion types from the OCC model, as well as "frustration" and "startle." The two main differences that occur in Em are intensity variables and cognition. Em uses only a subset of the intensity variables of the OCC model and the Affective Reasoner, resulting in a simpler model that was effective for all the characters required. As for cognition, Em focuses on a broad agent architecture that places less emphasis on strictly cognitive models of emotion. For example, Em emotion generators do not rely on complex cognitive modelling of other agents' goals to determine why an agent is behaving in some way. Instead, perceptual recognition of emotional states given some behaviour is built directly into a character. This simplifies emotion generation at the potential cost of loss of generality.

Once emotions have been generated, they are stored in a hierarchical data structure, illustrated in Figure 2.8. Each emotion type is represented as a node in a tree data structure organized by positive and negative emotions. A node in the hierarchy can hold multiple emotion structures (relating to different goals or situations) of one emotion type. The hierarchy provides for accessing emotions at both general and specific levels. The top level nodes (under "Total") are "Positive" and "Negative," which are useful for determining the general mood of an agent. When different emotion structures of the same type are stored together, they can be combined together to find an overall intensity for the emotion type. For example, an agent could be happy about finding $20 at the same time as being happy about his hockey team winning a game. In order to get a total intensity for the happiness of the agent, we need to combine the two emotion structures. Reilly proposes three different methods of combining emotion intensities: winner-takes-all, additive, and logarithmic. The winner-takes-all method uses the highest intensity of the group as the final intensity; this keeps the agent from having a strong reaction to many low-intensity emotions, but prevents

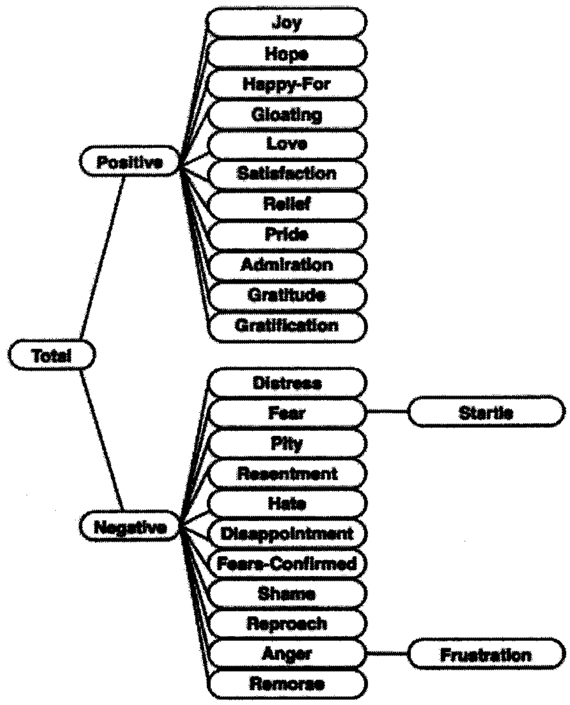

Figure 2.8: Default Em emotion type hierarchy [Rei96]

even a large number of medium-intensity emotions from causing a strong reaction. The additive method has the opposite problem: a few medium intensity emotions can cause a strong reaction, but so can a number of low-intensity emotions. The logarithmic approach is used in Em by default and retains the advantages of both the other methods, but is more complicated. Other algorithms in effect for emotion data structures are decay and queries. Emotions decay according to an explicit process defined for each emotion type. This explicit control is offered for artists to create more distinct personalities. Em provides a rich set of queries for accessing the emotional state. The queries are used primarily for generating behavioural features from emotional state.

The final step of the Em emotion process is expressing emotions. Emotional expression in Em is divided into two parts: mapping emotion structures to behavioural features, and using the features to express emotions by influencing other parts of the agent architecture. Behavioural features are general categories such as "aggressive" or "good-mood" that are generated and maintained using a mapping from emotion structures to behavioural features. The default emotion system in Em provides a large number of features meant to be coherent and straightforward while enabling freedom of artistic expression. A default behavioural feature map is given as a basis from which to develop unique characters. The feature map is written in scripting code and can be as simple or as complicated as the agent designer wishes. After behavioural features are generated, they have influence upon the agent's behaviour.

The Em architecture provides a number of avenues for emotional impact on behaviour aside from direct emotional expression. Emotions can influence several different parts of the Tok agent architecture, primarily including actions, natural language, and body state. We discuss each of these areas in turn, focusing on the influence on actions.

Emotions in Em can influence the action system by affecting the goals, behaviours, and actions of agents in several ways. By monitoring its emotional state for particular conditions, an agent can add new goals. For example, Bob could become angry at another agent and decide to get revenge. Emotions can influence the priority of goals, causing the agent to choose a different goal to pursue than it otherwise would. Emotions can also make it easier or harder for goals to succeed or fail by influencing the goals' criteria for success or failure. For example, an agent might have a goal to build a house of acceptable quality. If the agent is in a bad mood, it might not care about quality and have a lower standard, making it easier to achieve the goal. The importance of goals can be affected by the agent's emotional state, making the success or failure of the goal more or less important to the agent. For example, the more angry Bob gets, the more important his goal of revenge against the other agent would be.

The behaviours (plans) of an agent can be affected by emotions in four ways. The choice of behaviour for a goal can be influenced by emotional state preconditions on the behaviours. Similarly, emotions can cause an agent to switch from one behaviour to another due to emotional state context conditions on the behaviours. For example, Jack could be upset and decide to walk home from a party, but then calm down on the way and go back to get a ride from his friend. Emotions can also influence choices about what agent or object to use with a behaviour. For example, if Bill has a behaviour to call a friend, which friend he decides to call will depend on his emotional state. He would call Steve instead of Jack if he is angry at Jack. The number of times an agent will attempt a behaviour before giving up on it is also influenced by the agent's emotions.

In Em, actions can be viewed as simple behaviours or goals. In this way, they can be influenced by emotions in many of the same ways as behaviours and goals. In addition, emotions affect the style in which an action is performed. For example, an agent might have a generic action to walk across a room. If the agent is angry, he might stomp across the room. If he is happy, he might skip instead.

The natural language generation used by Em is quite straightforward, using a template-based system. While emotions could be used to influence natural language generation and understanding in a sophisticated way, Em uses them only to select between different templates that are written to show different emotions by word choice. The body state of agents in Em is influenced directly by emotions, and is used to output simple body appearance descriptions to the world. The mapping of behavioural features to body state is done using a standard set of rules. For example, when Sluggo is angry at another agent, he will have a red face and scowl at the other agent.

The believability of synthetic agents created with the Em architecture was validated with a moderately-sized user study, while the internals of the system were tested against a small user group. The main study involved 17 users who interacted with two versions of the Playground simulation – one with the emotional Melvin character, and another replacing Melvin with an Em-less version named Chuckie. After the interactions, the subjects were given questionnaires which asked them to rate each character on scales for emotionality, character, personality, and suspension of disbelief. The results indicated that Em characters were able to show emotion and that Em improves the overall quality of characters and their personalities. In the smaller study, three users were briefed on the operation of Em in the Robbery World simulation and given some sample traces that included emotion processing output. The subjects read through the traces and provided feedback on what they thought about the processing and how it might be improved. Generally, the users made suggestions about changing the details of the emotion system used for the agents. The main theme of the suggestions was that various aspects of emotions should be context-dependent. For example, emotional decay could be faster for emotions associated with certain goals. The robber's anger about being shot might decay slower than his anger about being insulted. Reilly concluded that the architecture can support most of the suggestions from the subjects, and so it can handle a wide variety of problems.

The Em emotion architecture is itself a significant contribution to the field as a working, powerful system. Reilly also shows that a purely cognitive model may not be sufficient to achieve the objective of believability, and strives to make the creation of believable synthetic agents accessible to artists. By affecting the goals and plans of agents, the Em architecture goes beyond the Affective Reasoner in terms of the influences that emotions can have on behaviour. One limitation of the system is the requirement for a significant amount of customized scripting to create a unique character. Another concern with the reliance on scripting is the difficulty of managing the code for several characters, each with a large amount of custom scripts. Defining emotion generators and behavioural feature maps with computer language scripts is useful for programmers, but not necessarily for artists. In this respect, the system does not go far enough to make itself approachable by most artists.

2.3.3 Plan-Based Approaches

An important contribution to computational models of emotion has been made by Jonathan Gratch, who created a generalized emotional model with AI planning concepts in the Émile system. His plan-based approach to appraisal is domain-independent, generating positive emotions if an agent believes a plan will succeed, and negative emotions if a plan is thwarted [Gra00]. This is in contrast to the approach of the Affective Reasoner, where agents use a large library of domain-specific construal frames to appraise situations in the world. If the agent encounters a new situation for which it does not have a specific construal frame, it is of no concern to the agent. The plan-based approach of Émile is designed to handle any situation because it does not depend on domain-specific frames.

Émile

The Émile system has five main processing stages [Gra00]. First, an agent builds and manipulates representations of plans to be used with a planner. It then qualitatively appraises events in the context of its plans and goals. A quantitative intensity value for the appraisals are generated, then used to generate emotional state. The appraisals are also used to inform action selection and planning. All of these stages are influenced by the plan representation.

The proven STRIPS planning representation is used in this system. Instead of appraising only single events, Émile also considers the state of plans in memory. This allows other factors besides external events to influence the appraisal process. For example, an agent could "sit and think" to develop its plans, which would cause new appraisals as the plans change. Since Émile appraises events using domain-independent rules about plans and goals, domain-specific information can be restricted to the planning operators. Goals in the system correspond to top level goals and sub-goals. Social standards are encoded as restraints on the plans, while preferences are represented as utility functions associated with goals (as opposed to objects). For example, the standard "one shouldn't damage public property" could be encoded as the constraint that the agent's plans may not contain actions that result in such damage. An example of a preference in Émile is a person having a goal of self-affirmation or mutual affection, instead of a simple encoding of "likes being near attractive people."

Emotion eliciting conditions (EEC) in Émile are generated from generic rules about plan features, rather than hard-coded domain-specific rules. Four conditions are highlighted by Gratch: *self* (whose perspective is being used to form the appraisal?), *desire-self* (is some characteristic desirable to the agent given in "self?"), *state* (what is the level of expectation associated with "desire-self?"), and *evaluation* (does the plan contain a praiseworthy or blameworthy act?). These EECs are used to determine what emotion structures arise from an appraisal. When multiple agents are interacting in the same world, their plans can conflict with one another. A standard that is present in Émile agents is that agents should not cause threats toward another friendly agent's plans. The planner detects if an agent's plan is a threat to the success of another agent's plans. This can lead to negative emotions directed toward the offending agent, depending on the qualitative intensity of the appraisal and the relationship between the agents.

Only two variables are used to determine the quantitative intensity value of appraisals: probability of goal attainment and goal importance. The probability of goal attainment is derived from a model based on the probability of an action actually having effect, and the probability of a sub-goal/precondition being satisfied. The goal importance depends on the reward from the goal and how much it helps to accomplish further goals. A further simplification of the model is that just five emotions are considered: hope, joy, fear, distress, and anger. Each of the emotions modelled by Émile use different functions to determine intensity based on the above factors, depending on the emotion. In order to avoid jarring changes in emotional state, emotional intensities are decayed similarly to the Velásquez Cathexis model [Vel97] (discussed in Section 2.3.4).

An example implementation involves two Émile agents with differing goals, personality, and knowledge. One agent (Steve) is rude and wants to go surfing, while the other (Jack) is nice and wants to make money. Their plans conflict with one another because there is a shared resource they each need, the car. The agents interact, observe, and communicate symbolically, generating appraisals and emotional state. A graphical user interface is provided for the researcher to observe the operation of the system, showing the plan structures, emotional state, communications (converted to English), and rudimentary facial expressions of the agents. Figure 2.9 shows the Émile interface with an exchange between Jack and Steve. An example exchange is given below.

> Jack: I want to make-some-big-money. *[Looks concerned, scratches his head, then, after devising a plan looks hopeful.]*
>
> Steve: I want to catch-some-waves. *[Looks concerned, scratches head, and*

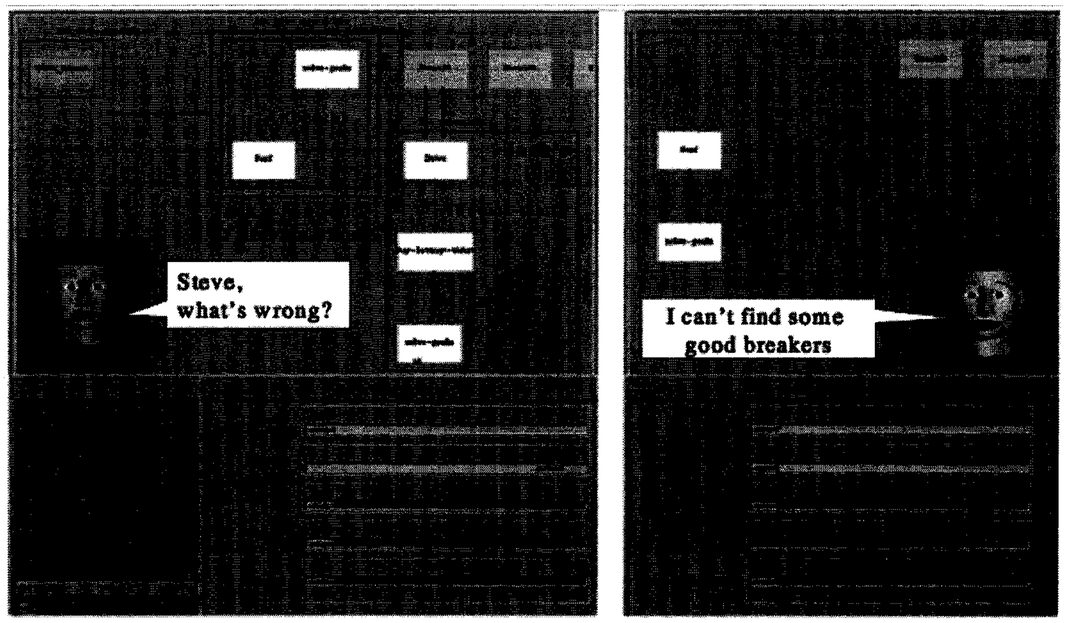

Figure 2.9: The Émile interface with two agents [Gra00]

continues to look concerned. Surfing is important to Steve and he cannot devise a plan.]

Jack: *[Perceives Steve's emotional state and generates an information request.]* Hey Steve, what's wrong?

Steve: *[Locates the appraisal generating the most intense negative emotional excitation. Communicates the associated plan fragment in a distressed tone of voice.]* I want to catch some waves but can't find any good breakers.

Jack: *[Incorporates Steve's plan fragment into plan memory and locates relevant information. Jack has knowledge of a wave report that establishes Steve's blocked subgoal]* Steve, does it help that someone did say theres some great waves near the pier?

Steve: *[Incorporates the communicated plan fragment. Completes a plan to go surfing and looks hopeful.]*

Jack: *[Perceives Steve's change in expression and seeks to confirm his expectation that the information he provided helped Steve.]* So that information helped?

Steve: *[Handles Jack's information request.]* Yes Jack. I plan to drive the car to the beach, then I plan to surf-my-brains-out.

Jack: *[Incorporates Steve's plan fragment and finds a conflict with his own plans. Based on personality, Jack attempts to negotiate a fair solution.]* Wait a second. Our plans conflict. I plan to drive the car to the quicky-mart then I plan to buy a-lottery-ticket.

Steve: *[Incorporates Jack's plan fragment and recognizes the same interaction. Based on personality model, Steve responds to interaction differently. Steve exits stage right.]* Later dude, I'm driving the car to the beach.

Jack: *[Perceives that car has departed without him. Looks angry. Says in angry voice:]* I want to kill-my-roommate. [Gra00]

Émile does not focus a great deal on the problem of using emotions to affect the planning algorithms directly, or for recognizing intentions, goals, or plans of other agents. Several of the methods used in Émile are intentionally simple. For instance, the agents communicate symbolic representations of plans that are easily recognized. The appraisal process is simplistic because it mainly considers concrete tasks and does not account for detailed interpersonal emotions or long term "life goals" that do not have a strong connection between goals and events. Even so, the system leverages a planner's general reasoning to reduce the complexity associated with the knowledge representation in earlier systems. Émile is an important contribution as the first general planning-based emotional computing system. By providing an example of a working domain-independent synthetic agent architecture, Gratch has led the research field away from the limitations of domain-specific construal theory.

Physical Expression

Further work by Marsella and Gratch concerns the problem of modelling emotions arising from the performance of a concrete task in a multi-agent simulation [MG01]. An extension to Émile focuses on the development of emotions arising from plan generation and the influence of emotions on physical expression such as gesture and body languages. The approach considers the relationship between events and an agent's goals and social standards. Domain-specific appraisal rules are avoided, so the only domain-specific information is gathered from the plan operator descriptions.

Non-verbal behaviours are very important in expressing emotion, but are hard to get right. Some behaviours are appropriate depending on the situation and what the character is saying, but others do not mix well. To generalize the problem, behaviour is based on

Figure 2.10: Interacting with the Mission Rehearsal Exercise [GM04]

a physical focus model [MJL00]. A character's attention is directed to different physical areas, driving how the character's behaviour is physicalized. Behaviours are grouped into modes based on emotional states, and types based on movements. At a given time, the agent will be in a certain mode depending upon its emotional state. Being in a mode predisposes the agent to using behaviours in the mode. By grouping behaviours into modes, the model keeps the agent's behaviour variable but consistent with its emotional state. In this model, three modes of physical focus are used: *body-focus*, *transitional*, and *communicative*. Depression and guilt are associated with body-focus, which is comprised of self-focused behaviour that avoids problem solving or communication. For example, an agent in the body-focus mode might avert her gaze from others or squeeze her forearm. The transitional mode includes behaviours that are more communicative and less depressed, such as fidgeting with objects or mumbled speech. This mode is associated with nervous emotions and indicates a burgeoning but suppressed willingness to interact with others. The third mode, communicative, occurs when an agent is fully willing to engage in problem solving and dialog with others. It is associated with positive emotions and is physically indicated by the appropriate use of gaze and the agent's full range of communicative abilities.

This model is used in a simulation called the Mission Rehearsal Exercise, a training scenario for military officers. Real time 3D graphics are used to provide immersion for the user and facilitate complex natural interaction with embodied virtual characters. The body language and gestures used by the agent are influenced by their emotional state using the physical focus model. In the Mission Rehearsal Exercise, a human user plays the role of a U.S. Army lieutenant in command of a platoon of soldiers. In the scenario, set in a peacekeeping environment, a traffic accident occurs involving military and civilian vehicles. A boy is injured in the accident and his mother awaits the arrival of the lieutenant. The lieutenant must deal with the situation, balancing several of his own mission goals against the events occurring in the simulation. Each of the other characters in the exercise are synthetic agents with their own goals and behaviours that do not necessarily agree with the lieutenant's. Figure 2.10 shows a user interacting with the system.

In this work, the authors speculate about exploring the idea that emotions could strongly affect the planning algorithm. Negative emotions would lead to narrow minded decision-making, while positive emotions would lead to broad problem solving that attempts to achieve multiple goals.

Coping Strategies

The impact of emotions on behaviour is an important part of any emotion modelling system. Marsella and Gratch further explore this facet of emotional modelling by applying coping strategies to Émile agents in the Mission Rehearsal Exercise [MG02]. Coping strategies or mechanisms are used by people to deal with their emotions, either reducing negative feelings or enhancing positive ones. Marsella and Gratch identify two broad categories of coping strategies. Problem-focused strategies involve acting externally on the world to change the situation that is causing the emotions, while emotion-focused strategies entail changing internal beliefs or focusing attention elsewhere. Emotion-focused strategies are more interesting from a modelling perspective, and that is where the authors direct their research.

Émile is extended once again in this research to support improved causal interpretation and focus of attention. Agents can appraise events from the perspective of other agents to model how another agent might feel about a situation. Focus of attention was implemented to solve the problem of multiple conflicting emotions, which could cause the characters in the system to appear frozen in a single emotion for some time. Attention is focused on the most intense emotion, which means that only the single focused emotion is used to influence appraisals and causal interpretations associated with the emotion.

The coping process contains three major steps. The first step is the occurrence of a focusing event, which causes the agent to focus on a relevant concern. For example, the agent could be asked a question or could notice a change in the environment. If the agent has a strong appraisal of the event in relation to its concerns, a coping elicitation frame (which gathers all related contextual and social information) is generated. Figure 2.11 shows an example of a coping elicitation frame from the sergeant in the Mission Rehearsal Exercise. The frame is generated when the lieutenant asks the sergeant what happened to cause the

Coping Frame:
Concern: Collision (mom.driver)
Emotion: Distress, 59.3
Focus-Event: Understand Speech
  Type: Information Request
  Agency: Superior
  Emotion: Distress, 40.3
Responsibility:
  Direct: Unspecified
  Indirect: Humvee Driver, Mom
  Potential: Sgt (superior of Humvee Driver)

Figure 2.11: Example of a coping elicitation frame [MG02]

accident. It summarizes the information about the event, the sergeant's beliefs about the lieutenant's feelings, and the responsibility for the associated concern. The coping elicitation frame is next matched to potential coping strategies. Several strategies are defined, depending on the personality of the agent, including both problem- and emotion-focused strategies. For the example in Figure 2.11, the sergeant has three possible coping strategies. He can try to solve the problem (make-amends) or take one of two emotion-focused strategies (assume-responsibility and shift-responsibility). Finally, behaviours are generated to take action on the selected strategy. Emotion-focused coping strategies can alter the belief structures of the agent, including assuming and shifting responsibility for actions attributed to agents. Agents may not take on clearly contradictory beliefs against what they directly perceive or are otherwise certain about. If a belief is uncertain, then it may be modified by coping and become incorrect nevertheless. Other behaviours available to agents affect body expression and the focus of attention. Changing the focus of attention can be thought of as distracting one's self. Body expression is linked with coping because the details of expressions differ widely between coping strategies and because the agent designer needs a good way to manage a large variety of available expressions.

2.3.4 Motivation-Based Behaviour

At the other end of the spectrum from high level cognitive models of emotion are systems that draw from low level physiological emotional research. While the OCC and other

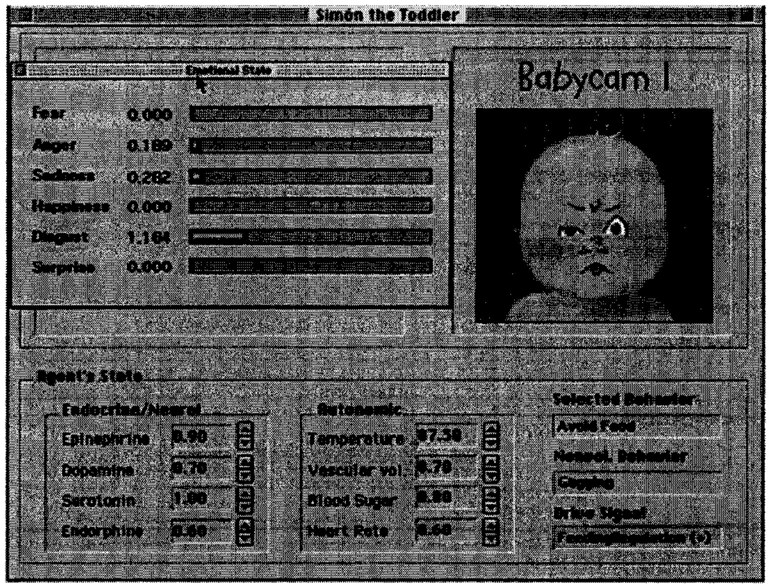

Figure 2.12: Interface to the Simón agent [Vel97]

cognitive emotion models can be considered "top down," generating emotions from cognitive thought processes, the Cathexis model proposed by Velásquez takes a "bottom up" approach by considering a wide variety of non-cognitive stimuli including brain chemistry, body senses, and internal physical drives or motivations. Models such as this are not directly relevant to our work, but are influential to the research field and provide important insights and ideas.

The Cathexis emotional model goes to a very detailed level and incorporates many factors driven by human biology [Vel97]. Six basic emotions are defined in the model: anger, fear, distress/sadness, enjoyment/happiness, disgust, and surprise. Each one is controlled by a proto-specialist (sub-agent) and has activation thresholds and a decay function. Factors that affect the generation of emotions include external sensors, internal sensors, and other proto-specialists. The factors include both cognitive and non-cognitive stimuli arranged into the categories of Neural (biochemical activity in the brain), Sensorimotor (information from the rest of the body, e.g. muscle positions), Motivational (pain regulation, emotions, and physical drives), and Cognitive (appraisals and interpretations of events, attributions, etc.). The model includes the expression of behaviour by means of action selection. As a whole the model is aimed at simulating the processes of emotion as experienced by humans in a neurophysical sense. While the emotional model has the potential to be complex, the first implementation (an agent modelling a young child named Simón) has a simple model of behaviour. This system is intended as a testbed for Cathexis, and as such includes an interface for the user to control the environment and the physical parameters of the agent. Simón has fifteen different behaviours to choose from and expresses them through a cartoon-like visual interface. Figure 2.12 shows a situation where Simón has just eaten *not-so-tasty* food and is expressing *disgust*.

Further investigation with Cathexis looks at how humans use intuition and emotional memories to help make good decisions [Vel98]. Velásquez investigates whether emotional memories can help an AI make better decisions. The idea is based on somatic markers that associate situations with good or bad outcomes in the past, and relate the past outcomes to current situations. In the improved version of the emotional model, cognitive factors are learned instead of being hard-coded. Additionally, the agent can have more than one behaviour activated at a time. The new model deals with primary emotions (instincts) and secondary emotions (cognitive, learned, and memory-based). Emotional memories are created whenever emotions are activated, linking stimuli and emotion. The next time the same stimuli is received, its associated emotional memory is activated and the emotion is "relived." For example, suppose Wendy is about to eat and the only food in the kitchen is a bad-tasting soup. Once she eats the soup, she might feel *disgust* because it has a pre-wired elicitor of foul tastes. At this point, an emotional memory is created by associating the primary *disgust* emotion with the stimulus that caused it (soup). The emotional memory has two effects, one behavioural and one emotional. When Wendy is hungry again, she will avoid eating if soup is the only available food, because of the learned aversion to soup. If she sees a bowl of soup, the emotional memory will activate and she will feel some measure of *disgust*. The idea of using emotional memories to influence behaviour delves further into deep emotional models that are based on what is postulated to happen in human emotional thought processing.

2.4 Emotional Influence on Behaviour

All of the synthetic agent systems discussed to this point involve some level of action, behaviour, or expression on the part of agents. Even so, most research in synthetic agents has focused on the generation of emotions, not the expression or effect of emotions. The OCC model, for instance, makes no attempt to cover emotional influence on behaviour. Most researchers have developed their own method of deciding behaviour, with varying results. The Affective Reasoner and the Em architecture both made a measure of effort to provide a layer of domain-independence for behaviour. The plan-based approaches of Gratch and

type parameterized action =	
(name:	STRING;
participants:	agent-and-objects;
applicability conditions:	BOOLEAN-expression;
preparatory specification:	*sequence* conditions-and-actions;
termination conditions:	BOOLEAN-expression;
post assertion:	STATEMENT;
during conditions:	STATEMENT;
purpose:	purpose-specification;
subactions:	par-constraint-graph;
parent action:	parameterized action;
previous action:	parameterized action;
concurrent action:	parameterized action;
next action:	parameterized action;
start:	time-specification;
duration:	time-specification;
priority:	INTEGER;
data:	ANY-TYPE;
kinematics:	kinematics-specification;
dynamics:	dynamics-specification;
manner:	manner-specification;
adverbs:	*sequence* adverb-specification;
failure:	failure-data).

Table 2.5: High level action PAR [AB02]

Marsella go further, providing generalized coping strategies for agents to respond to emotions. Two different approaches are outlined here which take advantage of research on non-verbal visual human behaviour, such as facial expressions, gestures, and body movement. We also discuss a third approach that focuses on the impact of emotions on cognitive decision-making.

2.4.1 Parametric Representations

A problem in the field of computational emotion models is that each researcher has come up with their own representation of rich personality and emotion models. Allbeck and Badler present a Parameterized Action Representation (PAR) that can be used for action, planning, reasoning, behaviours, and animation [AB02]. The purpose of PAR is to provide a representation that captures the semantics of human action and is extendable for future behaviours and animations. A basic high level PAR is an action that requires conditions to be satisfied, including activation conditions and termination conditions. The PAR contains the information necessary for an emotion architecture to select and execute an action. Table 2.5 shows the data structure for a high level action PAR. It includes, for instance, a variety of constraints and conditions for the action, links to related actions, and details about the execution of the action (from high level plans to low level details such as joint angle changes). PAR is intended to be sufficient to work with most emotion architectures and to

Space:	attention to the surroundings
Indirect:	flexible, meandering, wandering, multi-focus
Examples:	waving away bugs, slashing through plant growth
Direct:	single focus, channeled, undeviating
Examples:	pointing to a particular spot, threading a needle
Weight:	sense of the impact of ones movement
Light:	buoyant, delicate, easily overcoming gravity, marked by decreasing pressure
Examples:	dabbing paint on a canvas, describing the movement of a feather
Strong:	powerful, having an impact, increasing pressure into the movement
Examples:	punching, pushing a heavy object, expressing a firmly held opinion
Time:	lack or sense of urgency
Sustained:	lingering, leisurely, indulging in time
Examples:	stretching to yawn, stroking a pet
Sudden:	hurried, urgent
Examples:	swatting a fly, grabbing a child from the path of danger
Flow:	attitude towards bodily tension and control
Free:	uncontrolled, abandoned, unable to stop in the course of the movement
Examples:	waving wildly, shaking off water
Bound:	controlled, restrained, able to stop
Examples:	moving in slow motion, tai chi, carefully carrying a cup of hot liquid

Table 2.6: Motion factors of the Effort component [AB02]

be very comprehensive in terms of interaction with the environment. The representation incorporates the idea of how emotions affect the details of how physical behaviour is carried out. For example, a shy agent waves slowly, while an extroverted agent waves more quickly. A system called PARSYS stores PARs in a knowledge database and uses them to translate between natural language and character animation.

Several levels of action are modelled with PARs. Geometry, kinematics and physical action are handled by the system and information about these properties are stored in each PAR. The behavioural component is at a higher level and includes the agent's beliefs about the world. Agents are given goals and they try to complete their goals using PAR actions. The agents have access to stimuli from the world as well as information about failure states. If something goes wrong for an agent's action, it can detect the failure and try again (perhaps in a different way), or abort its action. The cognitive model allows for filtering and prioritizing of actions available to the agent for its plans. PAR is implemented in an emotion model which incorporates fixed traits and variable characteristics to form a dynamic personality. It is based on the OCEAN personality model. Personality and emotions affect decision-making and action selection.

The expressive system (EMOTE) is based on human movement observation research known as Laban Movement Analysis (LMA) [BL80]. The main components of LMA are Effort ("space, weight, time, and flow") and Shape ("changing forms that the body makes in space") [AB02]. Combining these factors with different weights and animations gives a wide range of movement for graphically animated characters. Table 2.6 gives a description of the Effort component with several examples. The goal of the EMOTE research is to link these motion factors with a general mapping to an emotional model based on OCC and OCEAN . The authors do not yet have this mapping, but intend to create a mapping between OCC and EMOTE using a learning process. Several demonstrations of EMOTE show how it maps the OCEAN traits to the factors space, weight, time, flow, and shape to create emotionally-influenced animations. One problem with a parametric representation is sensitivity. Rapid changes cause unnatural movement such as large instantaneous changes in joint angles. The authors' solution is to use a probability distribution over the parameters that is modulated by personality and mood.

2.4.2 Body Focus, Direction and Cinematography

Inspired by research on emotional coping strategies, Marsella et al. developed a system for agent-based interactive pedagogical drama that explores unique forms of expression [MJL00]. The agents have an emotional model that decides their actions, and special director and cinematographer agents decide the flow of the story and the expression/presentation. The user interface for the "learner" allows the user to direct the intentions of one of the characters; the character then acts on its own depending on its now-modified emotional state. The emotional model is primarily influenced by research on human stress and coping. The gestures available to the agents are also based on the idea of coping and revealing emotions. A dynamic script with dramatic and pedagogical goals determines the high level action, and fixed dialogue clips (recorded by voice actors) are used for verbal interaction. The script is decomposed hierarchically with increasing levels of dynamics, from three fixed acts at the top down to several alternate realizations of individual scenes. The director agent analyses the emotional states of agents and status of the goals to determine how the story moves forward; for example, selecting different strategies for learning certain concepts. For simplicity, the authors made one of the characters in the scenario the director.

Non-verbal behaviour is divided into four modes: strong body focus, body focus, transitional, and communicative. These are similar to later work by Marsella (described earlier in Section 2.3.3), where each mode has different tendencies of gestures that reveal something about the agent's emotions [MG01]. The transition between modes is caused by changes in emotional state. The graphical presentation uses an off-screen cinematography agent that weighs different factors in determining the presentation, including length of dialogue, strength of emotional display, and importance of particular lessons. Figure 2.13 shows the

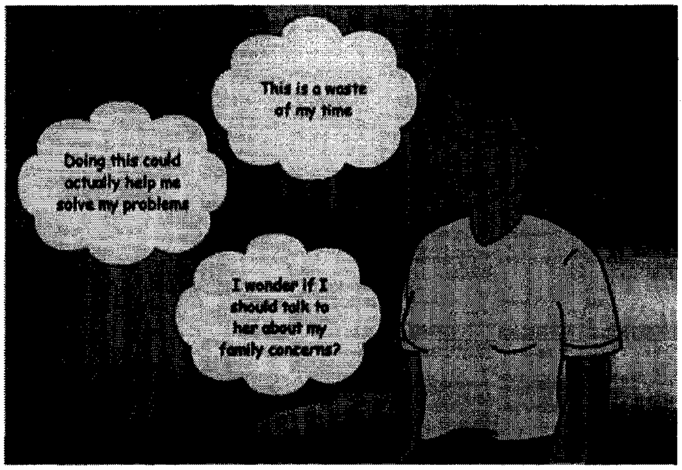

Figure 2.13: User interface with choice of thoughts for the agent [MJL00]

interface to the system, where the user can select a "thought" for the agent that influences her intentions.

2.4.3 Problem Solving

Belavkin investigated how emotional state might influence decision-making by affecting elements of a utility computation [Bel01]. He showed that a range of cognitive and emotional phenomena could be modelled using the ACT-R cognitive architecture [And93]. Belavkin was inspired by an earlier study which modelled the behaviour of adults and children in a problem solving task. The model was developed in ACT-R and produced results that matched the performance of adults. The authors found that a simple increase in the noise parameter of the conflict resolution algorithm resulted in an excellent match for the performance of children. Their result can be interpreted as meaning that children are more "noisy" or less rational than adults. Belavkin's work explores the idea that cognitive conflict resolution parameters have a strong relationship with emotions.

In the ACT-R architecture, conflict resolution is achieved by computing the *expected gain*, or utility, of each competing action for the agent. The expected gain considers the expected probability (P) that the action will achieve the goal, the value (G) of the goal, the expected cost (C) of the action, and a noise factor controlled by the *noise temperature* (τ). The action with the greatest expected gain is selected by the agent. An analysis of the expected gain variables indicates that the ratio G/τ represents the confidence of a problem solver, while the values of G and τ individually correspond to the "energy" of the cognitive process, also known as the *arousal* level of the agent.

In order to test the analysis of the expected gain computation, Belavkin models the

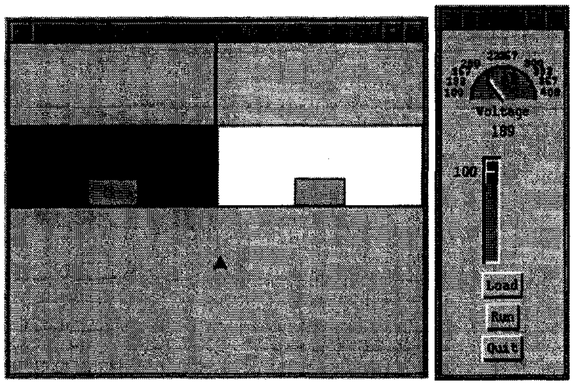

Figure 2.14: User interface of the "dancing mouse" experiment simulation [Bel01]

classical Yerkes-Dodsen "dancing mouse" experiment in ACT-R. The original experiment involved placing a mouse in front of two doors, one with a white card and the other with a black card. First, the mouse was trained to exit through either door. After two days, the situation was changed so that the mouse was only allowed to go through the white door, receiving a small electric shock if it attempted to use the black door. Subsequently, the experimenters randomly changed the order of the doors and measured how many mistakes the mouse made before forming a perfect habit (making no mistakes in three consecutive days). Yerkes and Dodsen explored how the speed of learning changed with respect to the contrast between the doors and the strength of the electric shock. They found that the best results occurred with medium levels of stimulation, forming an inverted-U function for learning performance with respect to stimulation level. Belavkin recreates this experimental setup by relating the contrast level to the G/τ ratio and relating the electric shock voltage indirectly to the goal value G. Figure 2.14 shows the user interface to the model, including a view of the mouse and the doors and a control panel for setting contrast and voltage. The mouse agent can sense the colour and positions of the doors, as well as electric shocks. It has actions to choose between the two doors and uses statistical probability learning. The results showed that decreasing the G/τ ratio caused probability learning performance to drop. Reducing noise and increasing goal value resulted in fewer errors. The model achieved similar results to the real experiment and verified the analysis of the expected gain variables.

Further analysis with problem solving models in ACT-R indicated that simply increasing noise did not satisfactorily reproduce the behaviour of 3-4 years old children. The model failed because it could not solve the problem and never abandoned the task. A solution to this problem is to vary the goal value and noise parameters of the model. Belavkin found that low values of G correspond to breadth-first search, giving priority to actions with low cost. On the other hand, high values of G correspond to depth-first search. Combining these observations with the effect of noise, Belavkin suggests that G (motivation) should increase gradually and τ should decrease gradually during problem solving. This corresponds with increasing confidence (G/τ) as the agent learns more about the problem. He also postulates that an emotional problem solver should associate positive emotions with increased motivation (G) and confidence (G/τ), while negative emotions should be accompanied by decreased motivation and confidence. This mode of behaviour corresponds closely with existing heuristic methods such as simulated annealing.

An important result of the ACT-R work is that emotions play an important role in problem solving. Previous experimental results with humans and animals were replicated using the ACT-R cognitive architecture by manipulating parameters that can be associated with emotional state. Belavkin also showed that emotions can make a positive impact on problem solving performance by approximating powerful heuristic methods.

2.4.4 Behaviour Under Stress

As part of their work to create a synthetic agent framework, Silverman et al. investigated the role of emotional and physiological stress on cognitive decision-making [SJW+02]. Their common mathematical framework (CMF) is intended to support a wide variety of emotional models. The architecture includes four subsystems: physiological, emotive, cognitive, and motor-expressive. The physiological subsystem responds to stimuli in the environment and contains physical processes or reservoirs such as energy, nutrients, and sleep. These processes are affected by stimuli to increase or decrease the stress level of the agent. Silverman models "integrated stress" (iSTRESS) which is composed of event stress (ES), time pressure (TP), and effective fatigue (EF). Event stress responds to positive and negative events that concern the agent. Time pressure is a ratio of the available vs. required time for the agent to complete its tasks. Effective fatigue is based on several physiological reservoirs. iSTRESS is calculated by combining these factors, modulated by the personality of the agent.

The iSTRESS value and its component parts are used to determine the overall coping style of the agent. The authors consider a classic inverted-U performance moderator function, similar to that which Belavkin modelled in ACT-R. The decision-making mode of an agent under stress is set according to certain thresholds. When the stress level is very low, the agent shows "unconflicted adherence to current action" and merely continues to do what it is already doing [SJW+02]. A somewhat higher amount of stress causes the agent to take a different action, but without carefully considering the situation. Medium levels of stress result in a vigilant decision-making mode, where the agent considers all available information and makes the best known choice. Increasing the stress brings the agent to near panic, where it will quickly choose from any action alternative without considering the consequences. Finally, very high levels of stress induce panic in the agent, causing it to either flee the situation or cower in place. These stress-driven decision style constraints dictate how the agent can behave in a cognitive fashion.

Each of the stress-induced decision-making modes influences emotional appraisal and action selection for the agent. For example, an agent under very little stress will not evaluate the situation using its emotional model and will ignore alternative action choices. On the other hand, an agent under extreme stress will only be concerned about its physiological state and either run away or cower in place, whichever seems safer. The motor and expressivity subsystems of the agent are also influenced by physiological state. The physiological subsystem imposes limitations on motor movement (e.g. an agent walks slower when fatigued).

The system was demonstrated with a prototype implementation that models crowd behaviour. The scenario involves a large number of agents in a protest situation, and has been used to test ideas for approaches to crowd control. Each agent in the system is fully realized and tries to achieve its goals while interacting with the other agents. In the demonstration scenario, a terrorist agent attempts to incite the crowd at a protest, potentially leading to rioting and looting. The authors detail how all of the behaviour of the agents is driven by both their goals and concerns and by their level of stress.

By focusing on the impact of physiological and emotional stress, Silverman et al. show that purely cognitive modes of decision-making are insufficient for many situations. Under conditions of very low or very high stress, people make decisions in entirely different ways. Synthetic agents that are affected by stress should behave more realistically in training scenarios and crowd simulations that involve stressful situations.

2.5 Scripting Approaches

Not all research on synthetic agents has focused on building general, psychologically-based, or physiologically-based personality and emotional models. A number of researchers have developed agent systems which produce believable synthetic agents without any complex underlying model. We group these systems together because they tend to involve a heavy use of customized scripting to achieve their goal of believable characters. Like the Em architecture, most script-based approaches are intended to make the creation of synthetic agents easy for non-technical content creators. We outline three systems: Improv, SCREAM/MPML, and a social-psychological model with an interesting expression system.

2.5.1 Improv

Like Reilly's Em architecture, the Improv system developed by Perlin and Goldberg focuses on the needs of the authors of interactive worlds [PG96]. Its fundamental goal is to allow authors to create believable characters that are consistent and interesting to users. The system has two main components, an Animation Engine and a Behaviour Engine. Improv provides a powerful scripting language that links the two components. The Animation Engine is based on the idea of degrees of freedom (DOF) in a body. The author specifies changes in DOF (actions), which are all blended together by the engine and displayed graphically. Actions can be related to one another to prevent conflicts. For example, if the agent is scratching his head with his left hand, he cannot touch his knee with his left hand at the same time. The Behaviour Engine can work either with deterministic parallel scripting of actions, or with probability models of behaviour that take values from the environment to select actions. For example, an agent could simply select randomly from a group of actions with a weighting determined by the author. Alternatively, the author can create decision rules to contextually influence the agent's choice. Decision rules take values from the environment to produce a weighting for potential actions. For example, a rule could select from going to a baseball game, a hockey game, or a football game based on the admission price, entertainment value, and sporting preferences of the agent. The authors present a plain English style scripting language for building action scripts and decision rules. Agents can co-ordinate together over a distributed network, requiring the system to overcome several issues involved in the distributed processing and display of agents. The system is designed so that each agent has one instance of the Behaviour Engine (one mind), but multiple instances of the Animation Engine (many bodies). Different users may see slightly different variations of actions by a given agent, but the basic behaviour of the agent will be the same. The authors assume that the agent designer will come up with his or her own emotional model for the agents. The behaviour engine of Improv allows you to have internal state, but does not include an emotional model by default.

2.5.2 SCREAM and MPML

Two systems introduced by Prendinger et al. focus on providing means for content authors (non-programmers) to create lifelike synthetic agents [PI02]. The systems include emotion generation and an animation engine to provide body movements/gestures in appropriate time with speech. Authors can use MPML (Multimodal Presentation Markup Language) scripting exclusively for fully scripted characters, or can use the SCREAM (SCRipting Emotion-based Agent Minds) engine to access the agent's emotional model. The SCREAM system follows the OCC emotion model, so that the significance of stimuli in regards to emotions is determined by emotion-eliciting conditions consisting of beliefs, goals, standards, and attitudes. These conditions are put into 22 emotion types or classes (e.g. joy, happy for, angry at). The content author combines the eliciting conditions together to make rules that determine how emotions are generated. Multiple emotions can be active in SCREAM, but only one is selected to display an expression. The emotions combine together to make a dominant mood. Characters can prefer to have positive or negative moods depending on their personalities. Emotions decay over time, depending on the emotion type and character "agreeability." Personality and standards are static, but goals, beliefs, attitudes, and social variables are dynamic.

The authors discuss the concept of emotion regulation, "a process that decides whether an emotion is expressed or suppressed" [PI02]. The decision is based on an agent's personality parameters and whether the agent sees the emotion expression as a threat to itself, among other factors. The animation engine for the system is implemented using the Microsoft Agent package, which includes 3D cartoon characters, text to speech, and speech recognition. MPML is used to script scenarios using characters embodied in the Microsoft Agent environment.

2.5.3 Social-Psychological Improvisation

A social-psychological model developed by Rousseau and Hayes-Roth involves improvisation of agent behaviour without detailed planning [RHR97a, RHR97b]. This model has components distinguishing personality traits, moods, and attitudes. In this research, personality traits are structures that specify the correlation between traits, moods, attitudes, and behavioural tendencies such as friendliness or confidence. Each trait has a value indicating its strength and valence, and traits can be defined using either simple values or complex conditional values involving many variables. For example, an agent might have a fixed confidence level of 6, but a friendliness level depending on how much she likes the other agents she is talking to. The concept of mood in this model is similar to emotional state in other models, but it is very coarse-grained. Moods are either self-directed or agent-directed, but are not associated with particular events or goals. For example, an agent could be generally happy, but also angry at a particular other agent. Attitudes determine how agents relate to one another socially. Attitudes used in this system include concepts such as liking and trusting. When the system is running, it determines what the social situation is and computes new values for attitudes, moods, and personality traits. The traits are then used to modulate the choice of behaviour for the agent. The high level action selection is simple, with no planning or goals; the agent has a single task to do. The behaviour selection chooses one of a few variations for a particular behaviour. For example, to walk from one place to another, an angry agent will stomp, while a lazy agent will stroll leisurely. The actions and variations available to an agent are domain-specific and predetermined by the agent author. Despite these limitations, the synthetic agents in this system are shown to exhibit recognizable personality and are considered believable by users.

2.6 System Evaluation

Most evaluations of computational models of emotion rely on the idea of believability of synthetic agents. The Em architecture, for example, was primarily evaluated on the basis of how it made characters more believable. Gratch and Marsella note that there is no generally agreed definition for believability, and that it does not necessarily have a correlation with realism. Instead of trying to subjectively determine believability, they turn to evaluating specific questions of functionality [GM04]. The EMA emotion processing model, based on the earlier Émile architecture, is evaluated in terms of these questions. The EMA model uses cognitive appraisal and decision-theoretic planning techniques, so that its plan representations include causal relationships between events, states, and beliefs. The coping strategies explored in Émile are made richer with improved support for modelling causality, attributions of credit or blame, and commitments to intentions and beliefs.

The first method of evaluation used with EMA measures how well the model captures how real people appraise and cope with changing situations. The authors selected a standard psychological questionnaire used to measure coping responses. The test involves presenting a subject with a stereotypical situation and asking questions about their appraisal and coping strategies. The subject is next questioned about how they would respond to changes in the situation. Responses from all phases of the test are scored in comparison to how "normal healthy adults" respond. For EMA, abstract situation conditions were encoded as causal structures and updated according to the standard test. The appraisals and coping strategies from the model were recorded and compared to normal human responses. Gratch and Marsella found that most of the trends illustrated by human responses were supported by the model. Using a standard psychological instrument as a measurement for the emotional model provides a relatively fair and direct test for questions about emotional dynamics. Two concerns, however, are that the scenarios were constructed abstractly by the researchers, and that the model's responses were compared with aggregate trends. There is a possibility that the aggregate trends may not be a good approximation of any particular individual, and so the model may not be able to reflect the responses of a unique real person.

The second study involved human subjects observing recordings of a Mission Rehearsal Exercise simulation. The purpose of the study was to determine whether non-verbal behaviours of synthetic agents influence the responses of human subjects to an ambiguous situation in the same way that non-verbal behaviours of other humans do (an effect called "social referencing"). The subjects viewed two versions of a scenario where the lieutenant must make a decision, advised by his sergeant (a synthetic agent). In one version of the scenario, the other soldiers (also synthetic agents) show head nods and expressions that express agreement with the sergeant and disagreement with the lieutenant, and in the second version their behaviours are reversed. The subjects were asked if they agreed with the sergeant, with the soldiers, and how confident they were about their decisions. The results showed that the non-verbal behaviours had a significant influence on the decisions. However, it was not clear that the influence was a result of social referencing, because the same effect might be achieved by adding the words "the sergeant is right" onto the screen. Despite this drawback, the evaluation is useful because it directly measures how human decision-making is influenced by emotional behaviours. This allows us to make more confident conclusions about the usefulness of emotional behaviour for synthetic agents.

The evaluation scheme devised by Gratch and Marsella avoids the problems associated with determining the "believability" of agents. Even so, the believability of synthetic agents is an important metric, ill-defined or not. Reilly's user study of Em, with several direct and indirect questions relating to strength of character and emotions, was an attempt to measure believability in an objective way [Rei96]. Similar approaches have been used by other researchers, including Rousseau and Hayes-Rothe for their script-based system [RHR97a] and Marsella et. al with their system for interactive pedagogical drama [MJL00]. It is clear that believability is a useful property for synthetic agents to have, and efforts to measure it objectively remain an active area of research.

2.7 Summary

In this chapter, we presented a broad view of the synthetic agent research field. We also gave some focus to a few landmark theories and implementations that have become key resources for researchers in this area. Taking in the area as a whole, we can get a grasp on the research techniques and approaches used for synthetic agents. Certain decisions are made in each case: what emotions will be modelled, how they are generated and updated, how (or if) they influence and interact with one another, and how they affect the behaviour of the agent. Synthetic agent systems are primarily concerned with the chain from input to output. The believability of an agent rests on this: given an agent and some input, does the output (behaviour of the agent) make sense? The answers depends greatly on the personality and emotional state of the agent, not to mention the variable interpretation of human observers.

Computational models of emotion tend to involve several issues of complexity. Human behaviour occurs on a wide range of levels, from chemical reactions in the brain and fine motor movement to abstract cognitive reasoning and long term life goals. Few models attempt to cover the entire range, most opting instead to select some slice and finesse the rest. For example, the OCC model operates at a very high cognitive level and relies on implementation details to provide behaviour even at a body movement level. A related aspect of complexity is the number and nature of influential variables in the system. Designers and implementers of synthetic agent systems must determine a set of variables that produce distinct and believable behaviour across different agents. Not only that, if the system is to be used by non-scientist scenario designers, it must neither be overwhelming nor frustratingly inflexible.

Chapter 3

Problem Statement

3.1 Overview

Our focus is primarily on how the emotional state of an agent influences its cognitive decision-making process. We explore how the "expression" of emotional state can be made through the kinds of decisions an agent makes in solving simple problems in a single-agent environment. This work is at the level of planning and deciding in a simulated world, and takes advantage of an existing planning architecture. Agents do not develop full plans by themselves, but make decisions deliberatively and reactively to select between alternative ways to achieve their goals. Our research on emotional influence focuses here, at the point that the agent decides between one available course of action or another. The main influence we consider is how emotional state affects the perception of the relative merits of plans. For example, a joyful agent might overestimate the likelihood that a plan will succeed, while a frustrated agent could disregard the drawbacks of a costly course of action.

In our system, the personality model is straightforward. Since we are focusing on the effect of emotional state on decision-making, our representation of personality focuses on the factors which are most related to emotional influence. Much of the workings of the system can be influenced by agent personality settings. The settings most likely to differ between agents are emotion activation thresholds, which determine how easily an agent will have an active emotional state, that in turn will affect the agent's decision-making. Higher threshold settings lead to less emotional influence on behaviour. Other personality settings are normally set to reasonable defaults based on the OCC model and other models, but can be changed to suit different characters. These factors affect both emotion generation and emotional influence on behaviour. Our personality model does not include representations of standards and preferences.

A central notion in our model is that of utility: the usefulness of a plan to achieve a goal. Every option or plan available to an agent has some utility perceived by the agent and influenced by the agent's emotional state. All of the applicable plans are in competition with one another, and the one with the highest utility is chosen by the agent. This leads to an important problem – for a given problem scenario, what is the correct utility for each plan? For certain domains an objective utility is relatively easily computed for each plan using search or some other method. However, the domains normally encountered by synthetic agents are not so straightforward. The utilities of plans in a scenario developed for artistic or pedagogical purposes are usually determined by an artist or designer to encourage the agent to behave in a certain way. These subjective utilities may not result in the agent solving a problem in the "optimal" fashion according to the usual standard. However, the agent behaviour is more believable or realistic, since humans rarely behave optimally. This issue will return when we consider the behaviour of "rational" agents against synthetic agents in scenarios involving subjective utility.

Another element of many emotional models involves the problem of getting rid of emotions. Once an agent has some emotional state, its behaviour is influenced by the active emotions. However, the emotional state cannot last forever; when a person is angry, his anger must be dissipated in one way or another. Emotional decay is one solution to this problem; the coping strategies explored by Gratch are another. We decided to explore the concept of homeostasis in the context of emotional state. Homeostasis is the tendency for systems, especially biological systems, to move toward a stable equilibrium. In our case, that means the tendency for the emotional state of an agent to move toward a neutral condition. When an agent in our model has active emotional state, it triggers a system marker to return itself to the neutral condition. Plans for the agent to achieve this goal are available to the agent and compete on the same decision-making playing field as the agent's "regular" plans for achieving its main goals. Joyfulness itself leads to an action, and sadness leads to a different action. The concept of homeostasis is useful for exploring the idea that an emotional state itself gives rise to behaviours in the world, behaviours that are not directed at the achievement of a particular external world state. Rather, they are aimed at the achievement of internal states.

3.2 Research Objectives

The work detailed in this thesis has three main inter-related research objectives. First, we explore how emotional state computations can be used to impact cognitive decision-making in simple problem solving situations. The effects we explore are domain-independent and relate purely to a goal-based planning architecture. Our second objective is to determine how emotional state gives rise to potentially complex plan-based behaviours in the world. Since these behaviours are implemented as plans, they are influenced by emotional state just like other plans. This kind of expression that competes with the agent's "regular" behaviour has not been explored elsewhere. Since we are dealing with synthetic agents, we are concerned with the problem of believability. Our third objective is to show that both elements (influence on decision-making and expressive plans) give rise to believable characters. We wish to determine the effect that each individual element, as well as the combination of elements, has on believability. In order to do this, we develop experiments that evaluate synthetic agents using these elements.

3.3 Evaluation

The platform we use to explore these ideas is a Java plan-based architecture called JAM [Hub99]. By using an existing, freely available architecture we take advantage of a rich plan representation and processing cycle. However, we also inherit a few constraints and problems within the architecture. We have overcome what limitations we encountered to build a useful agent system with emotional modelling. The main input to the system is a text-based scenario file that includes world state information and a set of plans available to the agent. For emotional modelling we require additional data in the scenario file as well as an agent personality specification file. It is possible to include interaction with the user, but we have only explored this to a limited extent (causing plans to succeed or fail, for example).

The text-based output of the system works in two modes. The first mode is used for system development as well as evaluation of system operation at an abstract or low level. In this mode, the system outputs which plans are chosen by the agent as well as the emotional state of the agent. This allows us to observe the exact behaviour of the agent and see why it behaves the way it does. The level of detail in this mode can be adjusted to see finer points of information such as utility computations, factors for emotional state generation, plan selection competition, and so on. The second mode of output is a simple narrative trace of decision-making and behaviour in a world where the agent must solve a problem. The system itself outputs narrative text describing domain-independent matters, while plans available to the agent include templates for scenario-specific narrative output. The result of running the system in this narrative mode is a "story" of what the agent thinks and does as the scenario plays out.

We test our system empirically in several ways, using each mode of output where appropriate. Our first evaluations use abstract structured scenarios to contrast the behaviour of "rational" agents with emotional agents. These tests both confirm the operation of the system and evaluate the patterns of decision-making and emotional state changes. The abstract tests do not show whether the emotional agents are behaving in a believable fashion, however. We developed a second experiment involving human subjects to measure how the core elements of our system affect how people perceive synthetic agents. A scenario based on a play script was used to generate narrative traces which were read by the subjects, who then answered a short questionnaire. In this experiment we test the influence of emotions versus personality as well as the effect of cognitive influence versus expressive plans.

3.4 Summary

In this chapter, we outlined the problem that we investigated and the directions we have taken to solve it. We take a plan-based approach to model a single synthetic agent working to accomplish a task. Our model draws from the OCC computational model of emotions, the plan-based synthetic agent systems pioneered by Gratch, and the cognitive decision-making work of Belavkin. Starting with an existing agent architecture, we apply these ideas in a novel way and add extensions of our own to form a working synthetic agent system. The system is a contribution to the frameworks of how emotional state impacts problem solving choices and decisions. We also developed a set of experiments to evaluate the system's performance in abstract problem solving and in a story-telling setting.

Chapter 4

System Foundation

4.1 Overview

Before we discuss our emotion model and architecture, we must first present the underlying planning architecture we use as the basis for our system. By building upon an existing system, we take advantage of a tried and tested platform with many useful features. In this chapter we provide a brief introduction to procedural reasoning systems as well as a broad treatment of the JAM architecture used in our work. We touch on important features of the architecture, give an example of system operation, and describe a few refinements we have made to JAM. It is important to understand how JAM works in order to see how we integrated our emotional model into the system. We present the details of the emotional model integration in the next chapter.

4.2 JAM

4.2.1 Procedural Reasoning Architectures

In dynamic environments, it is often desirable for software systems to have the ability to reason intelligently in real-time. SRI International developed a framework called the Procedural Reasoning System (PRS) that uses procedural knowledge from domain experts to achieve goals and tasks [Mye97]. Procedural knowledge is a representation of how to perform a set of actions to accomplish something. For example, a procedure to fulfill the task "clean the floor" might include the sequence of actions "put water in bucket," "put soap in bucket," "wet the mop," and "move mop over the floor." PRS itself is defined in a domain-independent way and provides a powerful language to express and execute procedural knowledge. Agents in PRS have beliefs about the world and attempt to achieve goals in the context of their beliefs. They usually have several ways to achieve a goal and

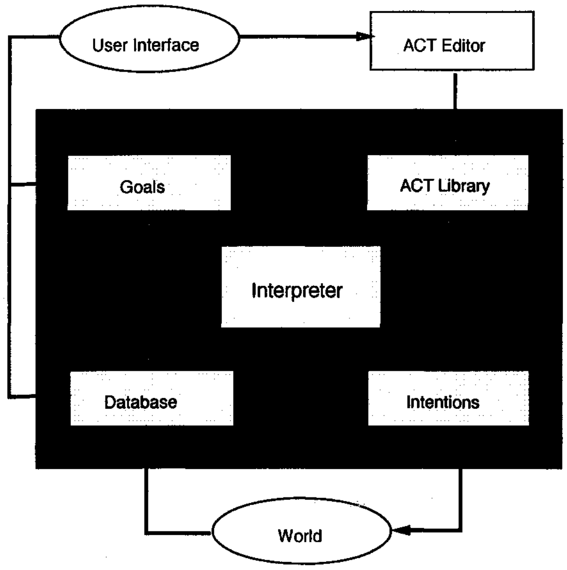

Figure 4.1: The PRS architecture [Mye97]

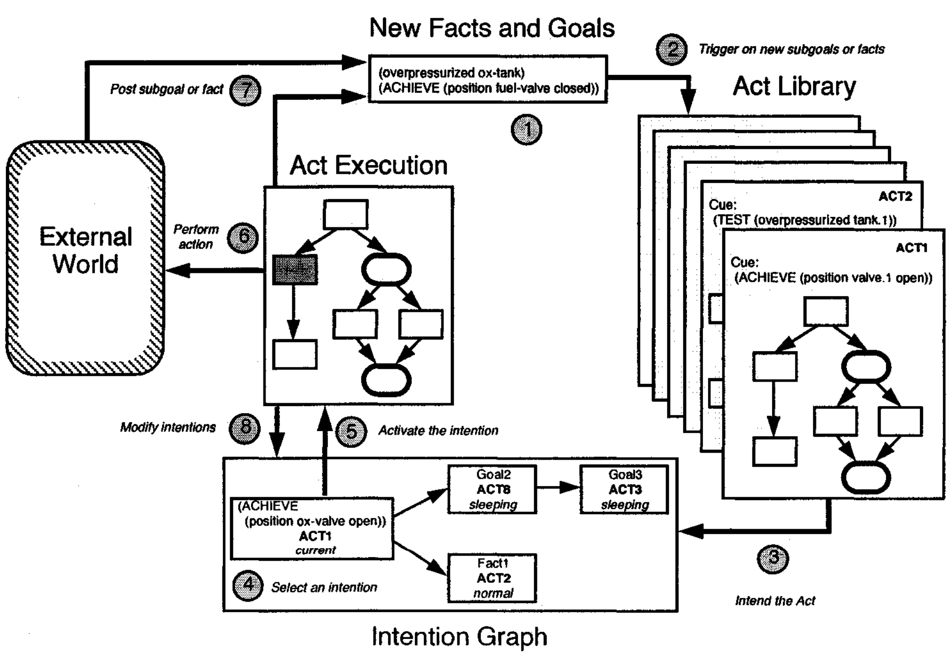

Figure 4.2: The PRS interpreter loop [Mye97]

select randomly between the procedures that are applicable to the goal. Agents are also able to react to new events happening in the world, integrating event-driven and goal-driven behaviours.

Many procedural reasoning architectures, including the one we use, are based on the ideas introduced in PRS. They tend to share a similar high level operational design consisting of four major data components. The original PRS has a *Database* with a representation of static and dynamic beliefs about the world, a set of *Goals* for the agent to achieve, a library of *Acts* (plans) containing procedural knowledge, and a representation of *Intentions* containing plans chosen for execution [Mye97]. Figure 4.1 shows the organization of the PRS architecture and how the different components interact. As we see in the figure, the central piece of the system is the *Interpreter*, which ties all the data components together in the execution loop.

Figure 4.2 shows the operation of the PRS interpreter loop. Changes in beliefs or goals cause the agent to consider the Acts in its library and intend one or more of them (selecting randomly). The agent may already have several Acts intended, so it selects one of them to execute. By default the agent selects randomly between its intended Acts, but it is possible to override this with a conflict resolution mechanism that orders intentions in some way (for example, according to importance). The conflict resolution mechanism in some PRS-style systems uses utility metrics to distinguish between goals and plans. It is also possible to build a conflict resolution mechanism called metalevel reasoning that uses Acts themselves as part of the reasoning process. We will see in Section 4.2.6 that the JAM architecture we extended for E-JAM takes advantage of both utility metrics and metalevel reasoning for its conflict resolution. The next action step in the selected intention is executed and may have some effect on the world, on beliefs or goals/sub-goals, or on existing intentions. At this point the process begins again, and continues until the agent achieves all of its goals or reaches some stopping condition. The short cycle of the interpreter loop allows PRS style agents to react quickly to changes in the world, update their beliefs, and decide on a different course of action.

The main strength of PRS style agents is their ability to use expert procedural knowledge in a dynamic fashion. Systems of this kind differ from traditional planning systems because they do not have the ability to construct new plans on the fly. In other words, PRS requires the agent to be pre-loaded with a set of plans to achieve its goals, and the agent can select only from those plans. It is not a traditional planning system, but an architecture for executing pre-defined sequences of procedural knowledge to achieve goals. From this point of view, the procedural knowledge approach is both a strength and a limitation. We chose to explore emotional modelling in this type of framework so that we could research plan-based approaches to synthetic agents without building a procedural reasoning agent architecture from scratch.

Another key feature of PRS is the ability of agents to keep track of multiple goals and intentions. Through the use of a customized conflict resolution mechanism, an agent can prioritize its intentions and direct its attention to important goals. This permits the agent to balance deliberative and reactive behaviour by the setting of priorities. For example, a PRS style agent can deliberate over the best ways to mop floors and build furniture, but it can also react immediately to an alarm presented by the environment. This can be achieved by giving a high priority to intentions that respond to the alarm, causing the agent to stop working on its other tasks and respond to the problem. While such a conflict resolution mechanism is not present by default, it can be added to PRS. The next architecture we discuss has a utility-based mechanism that achieves a balance of deliberative and reactive behaviour.

4.2.2 JAM Overview

JAM [Hub99] is an intelligent architecture based on several agent theories and frameworks, the most prominent of which are the Belief-Desires-Intention (BDI) theory and PRS planning framework. JAM is implemented in Java, so it runs on any system with an available Java Virtual Machine. There are four main components which concern us when extending JAM for emotional modelling. JAM's *World Model* (corresponding to the PRS *Database*) is a representation of the beliefs of an agent and is where all information external to the agent is stored. The *Intention Structure* is the model of the agent's goals and current intentions and the *Plan Library* is the collection of plans the agent can use to achieve those goals. Actual operation of the system is accomplished by the *Interpreter*, which selects and executes agent plans, and the *Observer*, which is used for cyclical functionality.

4.2.3 Beliefs

The World Model available in JAM is a global data store that represents the current state of the agent's world. Any type of information may be stored in the World Model. Each entry is a relation consisting of a base name and any number of parameters:

relation_name argument1 argument2 ... argumentN;

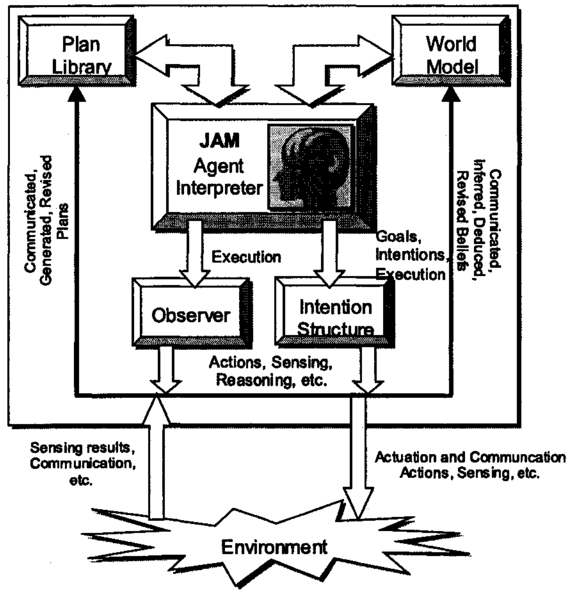

Figure 4.3: The JAM architecture [Hub99]

Arguments may be integers, floating point numbers, strings, or Java objects. The ordering and data types of arguments are unconstrained by JAM. The system does not directly support uncertainty about beliefs, but such functionality could be added by including and reasoning about additional world model relations.

An initial world state is specified in the Plan Library in the FACTS section. Additions, updates, and deletions to/from the World Model can be made inside any plan structure, including the Observer. Using the World Model is the only direct way for the agent to manipulate and act on information about its world.

4.2.4 Goals

The behaviour of a JAM agent is driven by the goals that it is directed to achieve. Each JAM goal is specified by a goal type, a relation label corresponding to the goal's name, any number of parameter arguments, and a utility value. Goals are stored in the agent's Intention Structure, which has a list of goals and associated intention threads (sub-goal chains). They are also represented in the World Model and may be used by plans like any other belief.

JAM supports four goal types. ACHIEVE goals are the most commonly used type of goals. The agent considers plans with matching ACHIEVE specifications to satisfy the goal. In addition, the agent monitors the World Model continuously so that if the specified relation becomes satisfied, the goal is achieved and plan execution stops. QUERY goals have the same function as ACHIEVE goals, but allow the agent designer to be more explicit about the purpose of the goal. For example, if an agent needs to find out the price of an item in a database, a goal specified as "QUERY price $item" is more natural and succinct to code than "ACHIEVE find-out price $item". PERFORM goals behave similarly to ACHIEVE goals, except that the World Model is not checked for premature goal success. With PERFORM goals, a plan is always selected and executed to achieve the goal. MAINTAIN goals function the same as ACHIEVE goals, except that a MAINTAIN goal is not removed from the goal list upon success. If the relation ever becomes unsatisfied, the agent must work again on the goal.

JAM defines two broad categories of goals: top-level goals and sub-goals. Top-level goals are usually specified in the GOALS section of the Plan Library. Goals added in this way can only contain constant parameters. Top-level goals can also be added by using the POST action within a plan. The POST action adds a single top-level goal (which can have variable parameters). In JAM, top-level goals are persistent. They remain on the agent's goal list until they are satisfied or they are removed using the UNPOST action. Sub-goals, which are created during plan execution, are by default not persistent. If a sub-goal fails, it is treated as a failure of the action in the plan and is not re-tried. As sub-goals are created, they are added to the intention thread for the associated top level goal.

4.2.5 Plans

In order to accomplish its goals, a JAM agent requires a library of plans that can be applied to those goals. A JAM plan contains both metadata about the plan as well as a procedural "program" for the agent to execute. Plan metadata includes the target (a goal or data-driven conclusion), plan name and documentation, precondition and/or ongoing context constraints, plan utility, and general attributes concerning plan characteristics. The plan body gives the procedure for executing the plan as a sequence of actions. Further actions can be specified in optional effects (executed upon plan success) or failure sections.

JAM provides many built-in actions for use in plans, as well as a method to execute primitive functions written in Java code (either JAM-provided or user-defined). A number of procedural programming constructs (conditionals, loops, etc.) are provided. Other control structures include posting new goals, abandoning existing goals, and executing actions in parallel threads. Figure 4.4 shows an example plan that demonstrates many of the available actions. Plan actions execute in sequence (or according to control structure actions) until the end of the encoded procedure, at which point the plan succeeds. A plan can prematurely succeed if the goal expression is satisfied during the execution of the plan. This

```
PLAN: {
    NAME: "Example plan"
    DOCUMENTATION: "This is a nonsensical plan"
    GOAL: ACHIEVE plan_example $distance;
    ATTRIBUTES: "test 1 cpu-use 3.0";
    PRECONDITION: (< $distance 50);
    CONTEXT:
        RETRIEVE task_complete $STATUS;
        (== $STATUS False);
    BODY:
        QUERY determine_task $task;
        FACT problem_solved $task $solved;
        OR {
            TEST (== $solved "YES");
            WAIT user_notified;
            RETRACT working_on_problem "True";
        } {
            TEST (== $solved "NO");
            ACHIEVE problem_decomposed;
            ATOMIC {
                ASSERT working_on_problem "True";
                MAINTAIN problem_decomposed;
            };
            ASSIGN $result (* 3 5);
        };
        UPDATE (task_complete) (task_complete "True");
    FAILURE:
        UPDATE (plan_example_failed) (plan_example_failed "True");
        EXECUTE print "Example failed.  Bailing out"
    EFFECTS:
        UPDATE (task_complete) (task_complete True);
}
```

Figure 4.4: Example JAM plan showing many of the available actions [Hub99]

means that the outside world can cause the plan to be rendered irrelevant, and the agent can recognize the achievement of the goal and stop working on it. If any action within a plan fails without an alternative control path to the completion of the plan procedure, the plan fails. The plan can also fail if the context constraint becomes invalid during execution.

4.2.6 System Operation

JAM agents operate through the Interpreter module. The Interpreter executes the following steps to produce agent behaviour:

1. Load goals, world state, and plans from the plan library file.

2. Repeat until all goals are achieved, or no goals remain:

   (a) Execute the Observer procedure.

   (b) Generate an Applicable Plan List (APL) for the Intention Structure.

   (c) If the APL is non-empty, choose and intend a plan from the APL.

   (d) Select an intended plan for execution.

   (e) Execute one step of the selected plan.

The Observer procedure provides support for processing that occurs on every cycle, regardless of what the agent is doing. The procedure is specified in the Plan Library as a sequence of plan actions. Any processing may be done in the Observer procedure except for sub-goaling. For example, the Observer could be used to implement a simple physics model to provide a physical world simulation for the agent. It can also be used to synchronize the JAM World Model with data from external sources, such as a graphical environment.

An Applicable Plan List (APL) is generated on every cycle by the Interpreter. The APL consists of all plans that can be intended for the agent's active goals (that do not already have a plan intended for them). Plans whose precondition or context constraints are not satisfied are not put on the APL. Thus the APL represents the set of all possible valid courses of action for the agent, given its current goals without intentions.

In step 2c of the Interpreter cycle, the agent must choose one plan from the APL to intend. Like PRS, JAM supports "metalevel" reasoning for plan selection, where the agent may have several ways to make the decision. Metalevel reasoning is achieved by writing plans in the Plan Library that target a special metalevel reasoning goal. These plans contain the same sort of operations as regular plans and compete on the same level with other plans when the agent makes decisions. Metalevel plans are used to select plans based

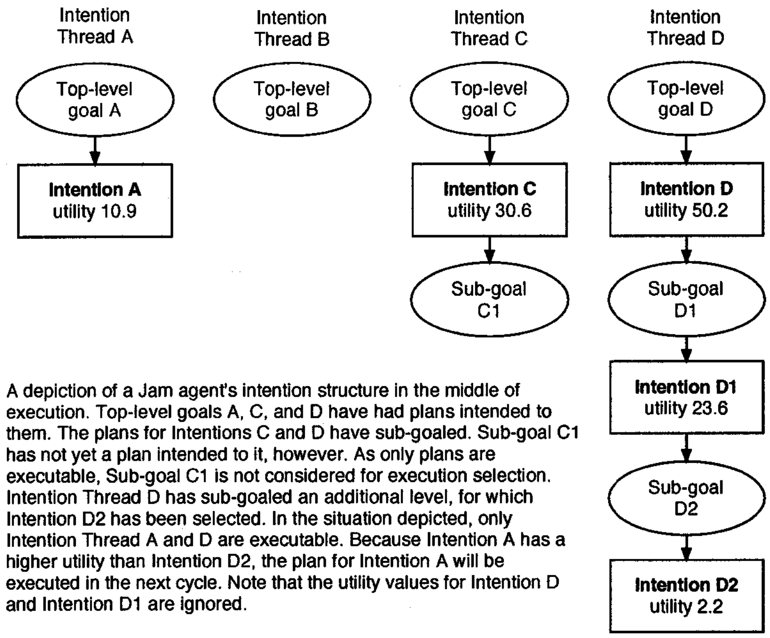

Figure 4.5: Example JAM Intention Structure state [Hub99]

on any sort of reasoning desired by the scenario designer. For example, the agent might always select the applicable plan with the lowest cost attribute value. Metalevel reasoning allows for a good balance of deliberative versus reactive behaviour because metalevel plans (deliberation) execute like any other plan, but can be suspended in favour of another (more important) plan in reaction to the environment. For our system we use JAM's default plan selection algorithm, where the agent sorts the APL by utility and selects randomly between the plans that share the highest utility. The agent *intends* the selected plan and places it with its associated goal in the Intention Structure. In other words, the agent makes a commitment to achieve the goal using the selected plan.

The next decision made in the Interpreter cycle is which plan to execute. There may be several goals with plans intended at the same time. JAM's default algorithm selects the leaf-goal intention with the highest utility for execution. The utility of intentions further up the same intention thread are ignored for the purpose of intention selection. When the agent selects an intention for a different goal than the currently executing goal, the current goal is suspended. The state of the intention thread with the suspended goal is stored, and will be resumed from the same state when the agent selects it for execution. If the context conditions of any part of the resumed intention thread are invalid upon resumption, plan failure will occur at the point of failure. The failure may "bubble up" all or part of the way to the top level goal. Figure 4.5 shows an example JAM Intention Structure for an agent in the middle of execution.

After selecting a plan to execute, all that remains for the Interpreter to do is execute the next action in the body of the selected plan. If the action results in plan success or failure, then that is resolved at this time by atomically executing the plan's success or failure effects section.

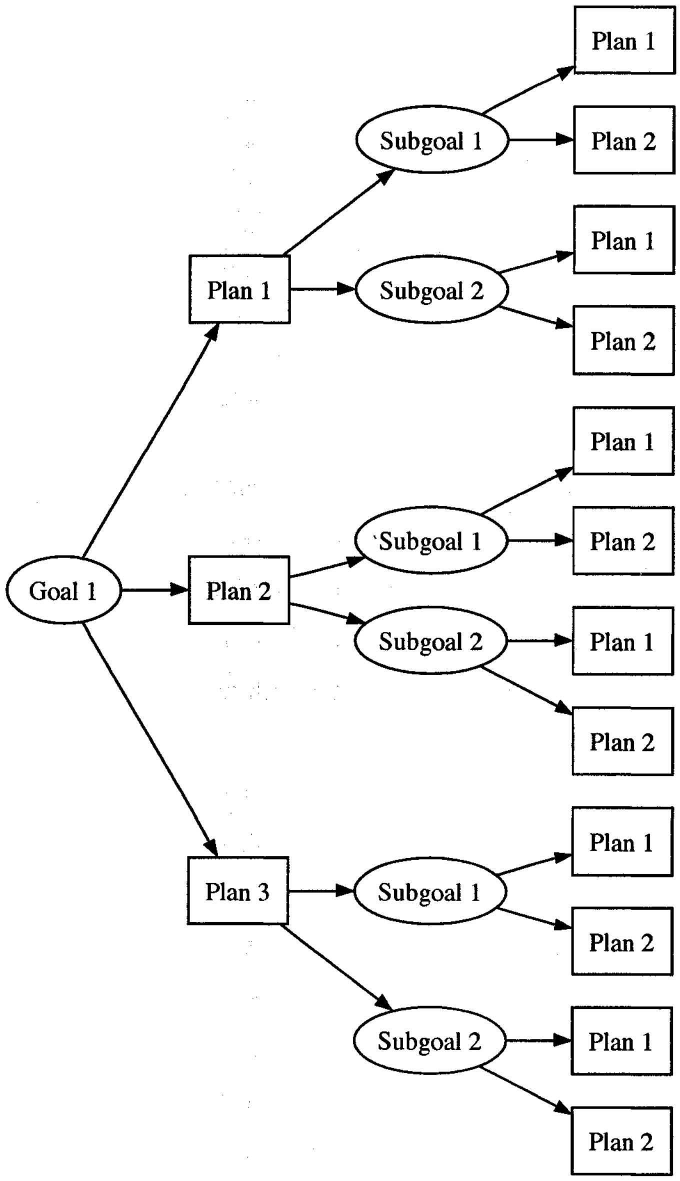

Figure 4.6: Graphical representation of an example JAM scenario. Ellipses represent goals and boxes represent plans.

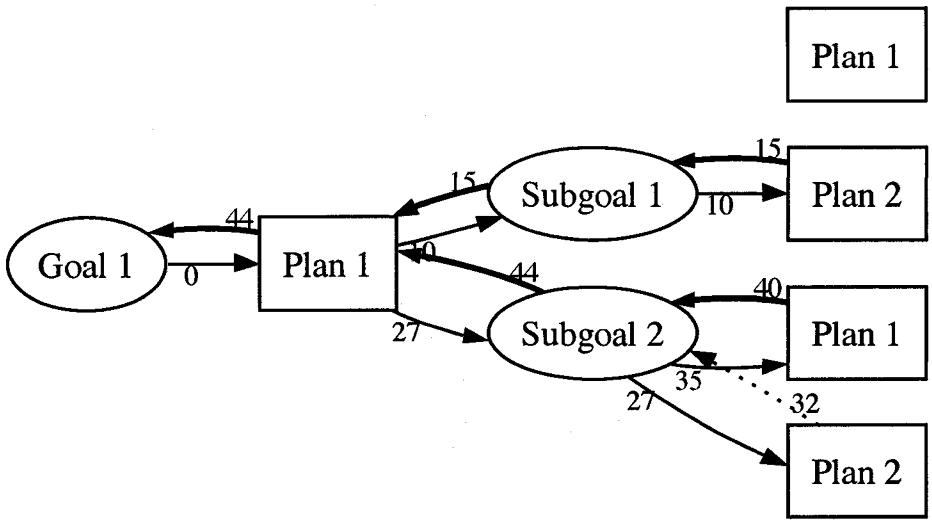

Figure 4.7: Graphical representation of agent behaviour in JAM. Ellipses represent goals and boxes represent plans. Edges from left to right indicate sub-goaling or plan selection. Bold edges and dotted edges from right to left indicate success and failure, respectively.

4.2.7 Example JAM Trace

Figure 4.6 shows a graphical representation of a simple JAM scenario. Goals are shown as ellipses and available plans are represented by boxes. The agent here has one top level goal, "Goal 1," which can be accomplished using any one of three plans. Each plan requires two sub-goals to be achieved by one of two plans each. However, each plan can only be attempted once. A similar version of this scenario is used as part of our experiments in a later chapter. Utility computations for the plans depend on utility variables defined in the plans and goals. These variables and computations will be discussed in the next chapter.

A graphical representation of the following annotated example trace is shown in Figure 4.7. The output trace shows how the JAM architecture operates in a cyclic fashion by considering the utility of applicable plans, selecting and intending plans, and executing plan actions. Cycles where the agent executes plan steps with no interesting output have been omitted for brevity. The figure, which is generated automatically from an expanded version of the trace output, shows the behaviour of the agent in a simplified form. The flow of decisions made by the agent is represented by directed edges labelled with the cycle number shortly after the decision was made. Arrows from left to right indicate sub-goaling or plan selection, bold arrows from right to left indicate plan/goal success, and dotted arrows from right to left indicate plan/goal failure. For example, on cycle 10, the agent initiates "Subgoal 1" and selects "Plan 2" to achieve it, both indicated by regular arrows. On cycle

15, the plan (and the goal) succeeds, indicated by a bold arrow, and the agent returns to the higher level plan.

```
JAM Parser Version 65 + 76i:
JAM definition parse successful.

Interpreter: starting cycle 0
Interpreter: Deciding on a plan
Computing utility for plan 'Goal 1 Plan 3'
  Original values: G =  0.50, P =  0.80, C =  0.80, N =  0.00
  Final utility = -0.40
Computing utility for plan 'Goal 1 Plan 2'
  Original values: G =  0.50, P =  0.50, C =  0.50, N =  0.00
  Final utility = -0.25
Computing utility for plan 'Goal 1 Plan 1'
  Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
  Final utility = -0.10
Interpreter: Selected plan "Goal 1 Plan 1" from APL.
Interpreter: Executing the intention structure.
IntentionStructure: Executing goal goal1
AGENT: Goal 1 Plan 1

Interpreter: starting cycle 1
Interpreter: Deciding on a plan
Interpreter: Executing something already in the intention structure.
IntentionStructure: Executing goal goal1
```

After loading the Plan Library, the Interpreter has a single goal for the agent, "Goal 1." There are three applicable plans for the goal, and "Plan 1" is selected because it has the highest utility. The utility is computed from several variables for each plan, but the important value for plan selection is the final utility. Once the plan is selected, the Intention Structure starts to execute the plan.

```
Interpreter: starting cycle 10
Interpreter: Deciding on a plan
Computing utility for plan 'Goal 1 Plan 1 Subgoal 1 Plan 2'
  Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
  Final utility = -0.10
Computing utility for plan 'Goal 1 Plan 1 Subgoal 1 Plan 1'
  Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
  Final utility = -0.10
Interpreter: Selected plan "Goal 1 Plan 1 Subgoal 1 Plan 2" from APL.
Interpreter: Executing the intention structure.
IntentionStructure: Executing goal goal1plan1subgoal1
AGENT: Goal 1 Plan 1 Subgoal 1 Plan 2

Interpreter: starting cycle 11
Interpreter: Deciding on a plan
Interpreter: Executing something already in the intention structure.
IntentionStructure: Executing goal goal1plan1subgoal1
```

After executing "Plan 1" for several cycles, the plan sub-goals to "Subgoal 1." As before, the Interpreter decides between the applicable plans for the goal. Since both applicable plans have the same utility, -0.10, the agent randomly selects between them. The intended plan is then executed by the Intention Structure.

```
Interpreter: starting cycle 15
Interpreter: Deciding on a plan
Interpreter: Executing something already in the intention structure.
```

```
44 IntentionStructure: Executing goal goal1plan1subgoal1
45 EVENT: Goal success: goal1plan1subgoal1
46
47 Interpreter: starting cycle 16
48 Interpreter: Deciding on a plan
49 Interpreter: Executing something already in the intention structure.
50 IntentionStructure: Executing goal goal1
51
52 ...
53
54 Interpreter: starting cycle 27
55 Interpreter: Deciding on a plan
56 Computing utility for plan 'Goal 1 Plan 1 Subgoal 2 Plan 2'
57   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
58   Final utility = -0.10
59 Computing utility for plan 'Goal 1 Plan 1 Subgoal 2 Plan 1'
60   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
61   Final utility = -0.10
62 Interpreter: Selected plan "Goal 1 Plan 1 Subgoal 2 Plan 2" from APL.
63 Interpreter: Executing the intention structure.
64 IntentionStructure: Executing goal goal1plan1subgoal2
65 AGENT: Goal 1 Plan 1 Subgoal 2 Plan 2
66
67 Interpreter: starting cycle 28
68 Interpreter: Deciding on a plan
69 Interpreter: Executing something already in the intention structure.
70 IntentionStructure: Executing goal goal1plan1subgoal2
```

On line 45 we see that the "Plan 1 Subgoal 1" succeeds. The agent returns to executing "Plan 1" for "Goal 1" and eventually comes to another sub-goal, "Subgoal 2." Again, the utility values for each applicable plan for "Subgoal 2" are the same, so one is randomly selected and executed.

```
71 Interpreter: starting cycle 32
72 Interpreter: Deciding on a plan
73 Interpreter: Executing something already in the intention structure.
74 IntentionStructure: Executing goal goal1plan1subgoal2
75 EVENT: Plan failure: Goal 1 Plan 1 Subgoal 2 Plan 2
76
77 Interpreter: starting cycle 33
78 Interpreter: Deciding on a plan
79 Interpreter: Executing something already in the intention structure.
80 IntentionStructure: Executing goal goal1
81
82 ...
83
84 Interpreter: starting cycle 35
85 Interpreter: Deciding on a plan
86 Computing utility for plan 'Goal 1 Plan 1 Subgoal 2 Plan 1'
87   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
88   Final utility = -0.10
89 Interpreter: Selected plan "Goal 1 Plan 1 Subgoal 2 Plan 1" from APL.
90 Interpreter: Executing the intention structure.
91 IntentionStructure: Executing goal goal1plan1subgoal2
92 AGENT: Goal 1 Plan 1 Subgoal 2 Plan 1
93
94 Interpreter: starting cycle 36
95 Interpreter: Deciding on a plan
96 Interpreter: Executing something already in the intention structure.
97 IntentionStructure: Executing goal goal1plan1subgoal2
```

As we can see on line 75, the agent experiences the failure of "Plan 2" for "Subgoal 2." As part of the modified sub-goal semantics discussed in Section 4.2.8 , the agent returns to the higher level "Plan 1" and tries to achieve "Subgoal 2" again. This time, the only applicable plan is "Plan 1" because it is only permitted to try each plan once. It selects the plan and begins to execute it.

```
Interpreter: starting cycle 40
Interpreter: Deciding on a plan
Interpreter: Executing something already in the intention structure.
IntentionStructure: Executing goal goal1plan1subgoal2
EVENT: Goal success: goal1plan1subgoal2

Interpreter: starting cycle 41
Interpreter: Deciding on a plan
Interpreter: Executing something already in the intention structure.
IntentionStructure: Executing goal goal1

...

Interpreter: starting cycle 44
Interpreter: Deciding on a plan
Interpreter: Executing something already in the intention structure.
IntentionStructure: Executing goal goal1
EVENT: Goal success: goal1

Interpreter: starting cycle 45
Interpreter: Deciding on a plan

JAM: All of the agent's top-level goals have been achieved!  Returning...
```

This time, the sub-goal succeeds and the agent returns to the higher level "Plan 1," which also succeeds. The end result is that the agent achieves "Goal 1" and is finished its work.

4.2.8 Modifications to JAM

As part of our investigation and development of an emotional model component with JAM, we made several modifications and enhancements to the system. Each enhancement was made in such a way as to preserve the functionality of JAM while improving support for designing agents which have actions in response to internal (emotional) state and which compute all plan utilities as a function of internal (emotional) state.

We explored the possibility of connecting a 3D graphical world (ANIMUS) to the JAM architecture, using JAM as the "brain" of an agent that existed in the 3D world [TB03]. The default runtime behaviour of JAM is not suited to such a situation because it is instantiated with a single call and runs by itself until it finishes. We modified the runtime behaviour of JAM so that it was possible to run a single cycle at a time. When JAM runs from the command line, it executes in a loop as before. However, with our modification it is now possible to control or manage the progress of JAM agents from an external programmatic perspective. This gives us a finer grain of control for the purposes of integrating an emotional model with JAM.

A significant problem was discovered in JAM's intention selection mechanism, which we were obliged to resolve. According to the design of JAM, the utility of intention stacks should correspond to the utility of the leaf (lowest level) intention on the stack. However, the implementation was using the utility of the top-level intention only. This results in incorrect behaviour with respect to the JAM documentation, even though it may be desirable for agents who are driven only by the importance of top level intentions. Additionally, this problem can lead (in rare cases) to a deadlock situation, causing the system to fall into an infinite loop condition. Our fix for the problem was simply to ensure JAM used the utility of the leaf intention in all cases. For example, consider the situation in Figure 4.5. The leaf utility for Intention Thread A is 10.9 and the leaf utility for Intention Thread D is 2.2, so the agent will select Thread A to execute. The utilities of Intention D (50.2) and D1 (23.6) are ignored because they are superseded by Intention D2.

Goal failure semantics in JAM are very different between top level goals (root desires) and sub-goals. When a plan fails for a top level goal, the Interpreter continues to try to achieve the goal by selecting another plan or the same plan again (any number of times). The sub-goal semantics, however, do not allow for repeated attempts at achieving a sub-goal. If a plan for a sub-goal fails, JAM considers the sub-goal to have failed, and so the plan that created the sub-goal also fails. For synthetic agents emulating human behaviour, it may be more realistic for an agent to try more than one time to achieve a goal, but without repeating plans (too often) after they fail. For example, an agent might have a goal to clean the floor, and decides to try a plan to achieve the goal by using a vacuum cleaner. A step in the plan is be a sub-goal to get a vacuum cleaner; ways to achieve this could include looking for one, buying one, borrowing one from a friend, etc. By default, a JAM agent will try a single plan to get the vacuum cleaner; if the plan fails, the agent will consider its goal to get a vacuum cleaner a failure. However, we would expect a person in this situation to normally try more than one plan before eventually giving up. This type of semantic allows both for repeated attempts to achieve a sub-goal and for goal failure (since retries are limited).

Rather than modifying JAM itself to support our desired semantic, we decided to specify plan language structures that would achieve the semantic. Each plan has an associated World Model relation indicating whether or not it has been tried before. When the plan is executed, it updates the World Model relation to reflect that it has been tried. Top level goals are monitored by the Observer, and if all plans are tried (and all fail), the goal is removed and a goal failure event is noted. Sub-goals are no longer as simple as a single plan ACHIEVE action. In order to sub-goal, a plan must include a loop that repeatedly tries to ACHIEVE the sub-goal until success or all known plans have been tried. If all plans are tried and fail, sub-goal failure is noted. The number of times a plan can be tried is controlled simply by a number which can be agent-specific. More complicated semantics could be developed in a similar way, but we found that our simple "try each plan at most once" semantic resulted in reasonable behaviour.

4.3 Summary

In this chapter, we gave an overview of procedural reasoning systems and the specific architecture that we used as the foundation for our synthetic agent system. The important features of procedural reasoning systems include the representation and use of expert procedural knowledge, multiple active intention threads, utility metrics for conflict resolution, and the balance of deliberative and reactive behaviour. PRS style agents can have many goals to pursue at one time and decide between a variety of plans to achieve their plans. The decision-making process can be simple or complex, but can be interrupted at any time to react to changes in the environment that are more important to the agent. This feature will prove crucial to the development of synthetic agents that can respond to an internal emotional state with reactive behaviour that switches away from ongoing cognitive deliberation and execution.

Chapter 5

E-JAM Architecture

5.1 Overview

This chapter deals with E-JAM, our extension to JAM that supports emotional modelling. We first present a high level perspective of the architecture, followed by details of the data structures and operation of the system.

As discussed in Chapter 3, we are focused on a single procedural reasoning agent that is given a set of goals to achieve in a simple world. The agent's emotional state is controlled solely by goal and plan related properties and events. The events that concern the agent for emotional state purposes include plan/goal success, goal failure, and plan failure. For example, the failure of an important goal might cause the agent to feel distress. Properties of the agent's goals and plans are used to update the agent's hope and fear about its goals. For instance, if the agent has several promising plans available to achieve a goal, it may feel hopeful about its prospect of success.

The emotional state of an agent in E-JAM influences its behaviour in two important ways. The first behavioural impact is on the utility metrics associated with plans and goals. In our framework, different emotions affect the utility computation in different ways. For example, a high level of distress might reduce the agent's perception of the probability of success of plans, leading it to select a different plan than it normally would. Thus, an agent might solve a problem in a different way when it is "frustrated" as opposed to when it is "relaxed." This means that emotional state affects how external goals assigned to the agent are achieved.

The second emotional influence on behaviour is that emotional states themselves cause the agent to create new goals for itself. These goals have the homeostatic purpose of returning the agent to a neutral emotional state, and they compete with externally assigned goals for the agent's attention. For example, an agent that is highly frustrated after repeatedly failing a task may decide to go take a walk, call a friend, or do some other frustration-reducing behaviour. The high intensity of frustration triggers an internal goal to reduce the emotional state by "expressing" the emotion through other behaviours. This goal competes with the other goals the agent already has, so even though the agent might wish to take a walk, a previous goal might be so important that the agent decides to work on it instead. In E-JAM, the accomplishment of an internal, emotionally-motivated goal reduces the emotional state of the agent and also causes the agent to turn its attention back to an externally assigned task.

To support the creation of internal homeostasis goals, E-JAM distinguishes between cognitive plans (C-Plans) and emotional plans (E-Plans). C-Plans are associated with external achievement or maintenance goals, as in the regular JAM architecture. E-Plans, on the other hand, are triggered by an extreme emotional state. Successful execution of E-Plans will reduce the emotional state of the agent and allow it to attend again to the externally assigned goals. Whether the agent is deciding between C-Plans or C-Plans and E-Plans, the utility of the plans is always affected by the agent's active emotional state. This means two things: that selection of C-Plans is impacted by emotional state regardless of whether the emotions are intense enough to trigger E-Plans; and that the utility computations for both C-Plans and E-Plans are influenced by emotional state. We are not arguing that the creation of "E-Goals" and execution of E-plans is conscious or deliberate on the part of the agent, as "C-Goals" and C-Plans might be regarded. Rather, we chose to use the existing mechanics of the architecture to design and implement a new construct for controlling agent behaviour.

Figure 5.1 shows an overview of how JAM has been extended to become E-JAM. The main addition to the JAM architecture is the Emotion Module, which contains representations of agent personality, emotional state, and intention trees. Other changes to the system are indicated by italicized elements highlighted in grey. The Plan Library now contains E-Plans and C-Plans, and all plans in the library have extra attributes that are used by the Emotion Module. The Intention Structure gives the Emotion Module information about goals and plans for use in prospect appraisal. This information is used to set up an additional representation of intention trees that includes persistent sub-goals. We require this redundant data structure because JAM itself does not keep track of sub-goals when plans for them fail. The Intention Structure uses the Emotion Module for emotionally-influenced utility computations and maintains a representation of emotional homeostasis goals and intentions just like other goals. The Interpreter informs the Emotion Module of goal and plan events for emotional appraisal, and the Emotion Module updates the World Model with a

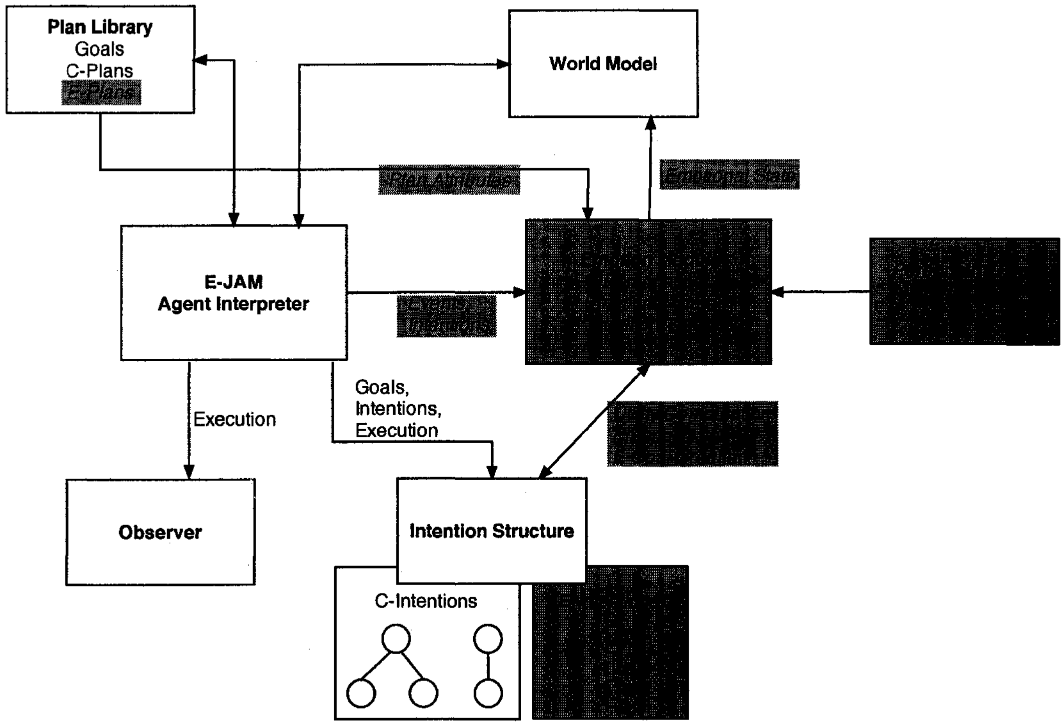

Figure 5.1: The E-JAM architecture. Changes from JAM are indicated by italicized elements highlighted in grey.

representation of the agent's emotional state for use at the plan level.

As in JAM, the Interpreter module is the main controller of the system. However, it calls into the Emotion Module at several key steps to trigger emotional appraisals and to compute plan utilities. The Interpreter executes the following steps to produce emotional agent behaviour. Differences from the JAM cycle are indicated with italics.

1. *Load agent personality files.*
    (a) *Initialize emotional state according to personality.*
2. Load goals, world state, C-Plans, *and E-Plans* from the plan library file.
3. Repeat until all goals are achieved *or deemed impossible (failed)*:
    (a) *Update the World Model with the current emotional state.*
    (b) Execute the Observer procedure, *which determines if any goals have succeeded or failed.*
    (c) Generate an Applicable Plan List (APL) for the Intention Structure.
    (d) *Appraise goal prospects using the APL and update the World Model.*

(e) If the APL is non-empty, choose and intend a plan from the APL *using the Emotion Module for utility computations.*

(f) Select an intended plan for execution *using the Emotion Module for utility computations.*

(g) Execute one step of the selected plan.

(h) *If a goal or plan related event occurs, appraise the event.*

The rest of the chapter will go into detail about the parts of the E-JAM architecture and the steps in the system cycle. Agent personality is represented in a fairly straightforward way, mainly using activation thresholds on emotion types to differentiate characters from one another. We initialize the starting values of emotions with values specified in the agent's personality file. E-JAM supports two main groups of emotions drawn from the OCC model, the Well-being emotions and the Prospect-based emotions. We organize the agent's emotions as pairs of opposites (for example, joy/distress). Next, we give the details of C-Plans and E-Plans before discussing emotional appraisal and behavioural influence. Emotional state is updated in two ways, from appraisals of events and from appraisals of prospects. While prospect appraisal happens every cycle, event appraisal only needs to happen whenever an event occurs. We describe in detail how emotions influence behaviour by affecting utility computations and by generating new internal homeostasis goals for E-Plans. Finally, we give an example trace from E-JAM that shows how all the pieces work together.

5.2 Definitions

Before we proceed further, several important concepts must be defined. The terminology of these definitions is mainly derived from existing research in the field, but we have created some new terms as well. The definitions of some existing terms have been narrowed in scope to fit the focus of our research.

In E-JAM, we consider only events related to goals and plans. Thus, an *event* is either a goal/plan success, goal failure, or plan failure. In JAM, when a plan succeeds, its associated goal also succeeds, so it is considered a single success event. Goal/plan success events can apply to either top level goals or sub-goals. A plan failure event occurs when a plan for a top level goal or for a sub-goal fails. This event is distinguished from goal failure, because an agent can often try another plan to achieve a goal. When an agent has no applicable plans left for a goal, a goal failure event occurs. This event can occur after an agent has exhausted all known plans, or if the world state changes so that previously applicable plans become unavailable due to unsatisfied prerequisites.

The OCC model makes reference to *desirable* and *undesirable* events. In E-JAM, we assume the agent desires to achieve all of its goals. Therefore, desirable events include goal/plan success events, and undesirable events include goal failure events and plan failure events. Another property of events is whether they are *expected,* or *prospected,* by the agent. In E-JAM we consider an event to be expected or prospected if the agent has strong emotions of hope or fear for the associated goal or plan. If the agent does not have strong hope or fear, the event is considered unexpected. This definition follows from the idea that if the agent does not have a strong belief about whether the goal or plan is likely to either succeed or fail, the actual end result will be unexpected. The status of an event can be either *unconfirmed, confirmed,* or *disconfirmed.* In E-JAM, all unconfirmed events are those which have not yet happened. Confirmed and disconfirmed events have either happened or "not happened," respectively, with absolute certainty. For example, when a plan is executing, its success or failure are both unconfirmed events. If the plan fails, its failure is confirmed and its success is disconfirmed.

Emotion types are defined in E-JAM the same way as in the OCC model. An emotion type is a representation of a kind of emotion, encompassing differences in intensity and subtle variations. For example, the emotion type "joy" subsumes the ideas of happiness, contentedness, ecstasy, etc. In E-JAM, emotion types are paired together by opposites to form *emotion dimensions.* An emotion dimension either has a single valenced emotion type, or two opposite valenced (positive and negative) emotion types. For example, the emotion dimension "hope/fear" has the positive emotion type hope and the negative emotion type fear. The actual emotional state of an agent contains instances of emotion dimensions, or *emotion instances.* An emotion instance has a numeric *value* which leads to an *intensity* depending on the value and the agent's personality. When an emotion instance's intensity is non-zero, we say the emotion instance is *active.* Emotion instances contain a *valenced emotion component* for each emotion type in the emotion dimension. We often refer to these components (in context) simply as emotions. For example, a hope/fear emotion instance has a positive component for hope (a hope emotion) and a negative component for fear (a fear emotion).

Most emotion instances are considered *global* and can have an influence on the utility computations for all the agent's goals. Hope/fear emotion instances, however, influence the utility computations only for a single goal. These are called *local* emotion instances. We discuss how local and global emotion instances are stored in Section 5.5.

The concept of *utility* is very important in E-JAM. The agent uses utility, a measure of the usefulness of a plan, to decide between different plans for intention and execution. In E-JAM we use four components or *variables* to determine utility: *goal value*, which is the importance of a goal to the agent; *probability of success*, which is the likelihood of a plan achieving its goal; *cost*, the amount of effort required by a plan; and *noise*, a random variable in the computation.

Original utility variables are specified by the scenario designer or are randomly generated, and the influence of emotional state leads to the computation of *perceived* utility variables. This distinction is very important for the discussion of emotional influence on utility.

5.3 Agent Personality

All E-JAM agents have a personality which influences both how their emotions are generated and how emotional state influences their behaviour. The personality is specified by the researcher and loaded by the system to set up the agent. In an effort to simplify the agent creation process, we provide a basic personality setup that includes default parameters based on the OCC model and other research, including emotional modelling research and psychological studies. When using these reasonable defaults, the main differences between agents become parameters for the initial emotional state and for activation thresholds. We follow this style of personality definition for our own experiments, discussed in later chapters.

Most personality parameters in our model apply to particular emotion dimensions. Figure 5.2 shows an example emotion dimension for "joy/distress" that is filled in with personality parameters. The joy/distress emotion dimension includes several personality parameters, along with a positive emotion type "joy" and negative emotion type "distress," each with their own set of different personality parameters.

The initial emotional state of an agent is set up in its personality definition by specifying an initial value for each emotion dimension. When starting the system, the agent's emotional state is loaded with these initial values and immediately begins to influence behaviour. Activation thresholds are the most important part of the personality specification. A threshold can be seen as the "breaking point" for the agent with respect to a particular emotion type. When the value of an emotion instance is below the threshold, the emotion instance has zero intensity and does not have influence on the behaviour of the agent (in

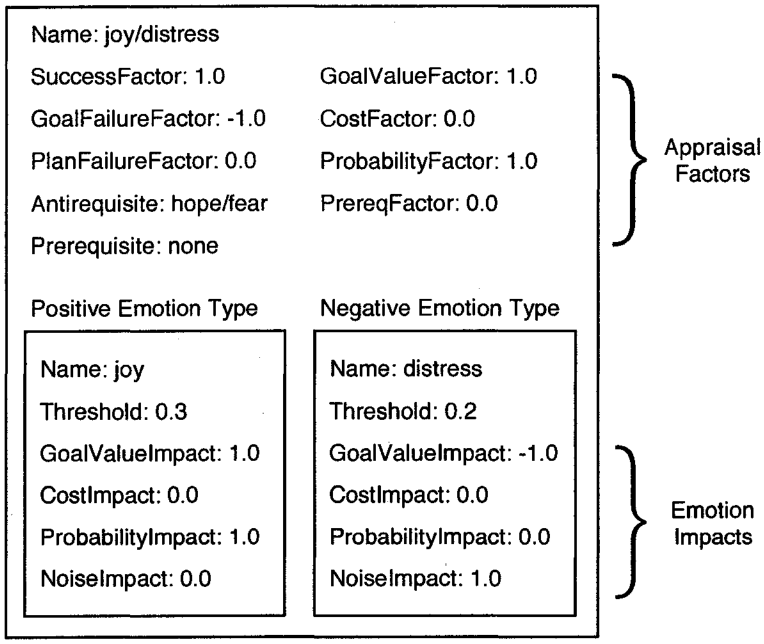

Figure 5.2: Emotion dimension data structure for joy/distress

terms of either utility or E-Plans). When the value is above the threshold, the emotion instance has non-zero intensity and affects behaviour by influencing utility and triggering E-Plans. Activation thresholds are a way to control the sensitivity of agents to particular emotions. For example, a "happy-go-lucky" agent might have a low threshold for joy, causing any small success to produce non-zero intensity for joy. On the other hand, a "grumpy" agent might have a high threshold for joy, requiring a number of success events to influence its utility computation using joy or to trigger a joy-related E-Plan. Figure 5.2 shows an example with a threshold of 0.3 for joy and a threshold of 0.2 for distress. In this example, the joy/distress emotion dimension is active when its value is greater than 0.3 or when its value is less than -0.2.

The other personality parameters are defined in the basic personality setup, but can be overridden by any particular agent personality specification. Each emotion dimension has parameters called *appraisal factors* that determine how events affect the emotional state of the agent. Appraisal factors can pertain to the events of goal/plan success, goal failure, or plan failure. For example, the value of joy/distress is increased on goal/plan success and decreased on goal failure. This effect is achieved by setting the goal/plan success factor to 1.0 and the goal failure factor to -1.0 (see Figure 5.2, top). Appraisal factors can have any real value, but we use only values of -1.0, 0.0, and 1.0 in our basic personality setup. We do this for simplicity and because the basic personality setup uses appraisal factors that result

Variable	Increased By	Decreased By
Goal Value	hope joy	fear distress
Probability of Success	joy satisfaction relief	fears-confirmed
Cost	disappointment	frustration
Noise	distress frustration	satisfaction

Table 5.1: Summary of emotional impacts on utility variables

in emotion appraisal similar to the OCC model.

Our personality model includes another set of appraisal factors which have similar characteristics to intensity variables in the OCC model. These appraisal factors determine how the contextual details of an event are used in emotional appraisal. For example, the higher the value of a goal that succeeds or fails, the higher the change in value of joy/distress on appraisal of the success or failure event. Prerequisite and anti-requisite factors are used to specify whether the emotion dimension requires another emotion dimension to be active for appraisal to occur. For example, joy/distress cannot occur when hope/fear is active (see Figure 5.2, top), but satisfaction/disappointment requires hope/fear to be active for its appraisal to happen. Again, appraisal factors can have any numeric value, but our basic personality setup uses values that follow the appraisal specification of the OCC model. Section 5.7 gives details of the emotion appraisal process and how these parameters come into play.

Emotion impacts are personality parameters which determine how active emotion instances influence the behaviour of the agent (see Figure 5.2, bottom). Each emotion type has emotion impact parameters that affect how the agent perceives utility variables such as goal value, plan cost, plan probability of success, and noise. For example, a joyful agent might perceive goal value and probability of success variables as higher than they really are. The original values of the utility variables are not changed, just the agent's perception of them. Section 5.8 describes in detail how the utility variables are affected by emotion impacts.

While the OCC model thoroughly specifies how emotions are generated (providing reasonable defaults for appraisal factors and variables), it is silent on the topic of emotional influence on behaviour. For the emotion impacts in our basic personality setup, we drew from a number of other sources in emotion research. Each source is noted along with the emotion dimensions in the following section. However, we did not find prior research to suggest the default settings for some of the emotion impact parameters. In these cases, we determined a reasonable setting for a plausible neutral agent. Table 5.1 summarizes the emotion impacts defined in our basic personality setup. Note, however, that an agent designer can define qualitatively different agents by changing which emotions impact which utility components (and by how much).

The last set of personality parameters are those for controlling the *noise* utility variable. Noise is generated as part of the utility computation in E-JAM, and is included because it is part of the ACT-R utility equation. The available noise parameters are mean and variance for Gaussian noise. These parameters affect all utility computations and are not associated with any particular emotion dimension.

5.4 Emotion Dimensions

Since our research focuses on single cognitive agents with event-driven behaviour, we model a subset of emotions from the OCC model including the Well-being emotions and the Prospect-based emotions. The OCC model does not include frustration, but we model it here as a separate emotion type. This section describes how we organized the emotion types into emotion dimensions and gives some details about the basic personality setup for the emotion dimensions.

In this section we refer to "hoped-for" and "feared-for" goals. A *hoped-for* goal is a goal which has an associated active hope/fear emotion dimension with a positive intensity. The event associated with the "hope" is the success of the goal. A *feared-for* goal is a goal which has an associated active hope/fear emotion dimension with a negative intensity. The event associated with the "fear" is the failure of the goal.

5.4.1 Well-being Emotions

The Well-being emotions consist of two emotion types, "joy" and "distress." These emotion types are opposite to one another on an intuitive level. A consideration of their definitions in the OCC model also justifies their placement in the same emotion dimension. The joy/distress emotion dimension is affected by goal success and goal failure events. To distinguish the Well-being emotions from the Prospect-based emotions in our model, they are only affected when the events are not associated with a hoped-for or feared-for goal. This distinction gives the Well-being emotions an element of surprise or unexpectedness. In our model we view the Well-being emotions as having minimal distinction from the vague ideas of "positive" and "negative" emotions. Their effects on behaviour tend to be

	Event	
hope/fear status	Goal Success	Goal Failure
hoped-for goal	satisfaction	disappointment
feared-for goal	relief	fears-confirmed

Table 5.2: Appraisal of E-JAM Prospect-based emotions

broad and excessive. Joy is a somewhat "raw" positive emotion that causes the agent to overestimate probability of success [NITD96]. Nygren showed that subjects with positive affect (activated by an unexpected reward) overestimated probability of success. Joy also causes agents to be more motivated to pursue their goals [Bel01]. Ddistress is an unrefined negative emotion that causes a lack of motivation, as well as erratic and poorly justifiable behaviour on the part of the agent [Bel01]. The translation of these effects into emotion impacts is that joy increases goal value and probability of success for all goals, and distress decreases goal value and increases noise for all goals (see Table 5.1).

5.4.2 Prospect-based Emotions

As the name suggests, Prospect-based emotions are predicated on specific future events and their expected outcomes. In this category, there are six emotion types organized into three emotion dimensions. The most important emotion dimension for this category is hope/fear. This emotion dimension is different from all the others because the agent can have multiple instances of it, one for each current goal. Additionally, the value of hope/fear emotion instances is not determined by appraisal of events, but by a separate appraisal process for prospected events. By definition, hope/fear is associated with unconfirmed prospective events. In E-JAM we can determine the probability that a goal will succeed or fail depending on the probability of success of each of the available plans for the goal. When a goal appears likely to succeed, hope/fear for that goal is increased. We assume that a hopeful agent wants to keep working on a particular goal to see it succeed. When a goal appears likely to fail, hope/fear for that goal is decreased. We assume that a fearful agent would lose motivation to pursue this goal if it seems unlikely to succeed.

The impact on utility metrics by hope/fear is summarized in Table 5.1. The perceived goal value for the associated goal is increased by hope and decreased by fear. For example, if an agent has several promising plans to achieve a goal, he will have hope for the goal and be more likely to work on it (due to the increased goal value). However, if plans for the goal become inapplicable or fail, his hope will decrease and perhaps turn into fear.

The other Prospect-based emotion dimensions are relief/fears-confirmed and satisfaction/disappointment. One may recall that these Prospect-based emotion types were organized differently in the OCC model. That model grouped the emotion types by confirmation or disconfirmation of a prospected event, while we group the emotion types by whether the associated goal is hoped-for or feared-for. Table 5.2 summarizes the organization of E-JAM Prospect-based emotions for appraisal purposes. The positive emotion in each emotion dimension corresponds to goal success, while the negative emotion corresponds to goal failure. At the appraisal of any given event, the associated goal can be hoped-for or feared-for, making it clear which emotion dimension and emotion type is involved. For example, if an agent has hope that he will achieve his goal to get money, he will experience disappointment if his goal fails. However, if his goal succeeds, he will experience satisfaction.

The default impacts on utility of the emotion dimensions relief/fears-confirmed and satisfaction/disappointment are meant to reflect a default reaction. We could not locate a reliable source detailing the effects of these emotions on behaviour. As shown in Table 5.1, a satisfied agent is more confident about its ability to achieve other goals because it had hope for a goal that succeeded. The agent perceives increased probability and decreased noise. A disappointed agent had hope for a goal but it failed despite all attempts. The agent will still want to pursue other goals, but with more concern about the cost of plans. The impact of relief is to increase the perception of probability of success because the agent's fears appear unwarranted. The fears-confirmed emotion leads the agent to behave pessimistically, decreasing the perception of probability of success, because its fears were correctly surmised.

5.4.3 Frustration

The idea of frustration as an emotion type is fairly common in the literature [Rei96, Bel01, Hud04]. It is not present in the OCC model because it does not relate directly to goal success or failure. Frustration arises when an agent's plan fails, even though the goal may still be achieved (by trying again, or by trying a different plan). Reilly implemented frustration in his architecture as a separate emotion type [Rei96], and here we do the same. The frustration emotion dimension contains the frustration emotion type as its negative component. The intensity of frustration in our model depends upon the importance of the associated goal and the cost of the associated plan. The more important the goal and the more costly the plan, the greater the frustration experienced when plans for the goal fail. When a goal succeeds, frustration is decreased. Frustration affects the utility computations for all of the agents' plans by decreasing perceived cost and increasing the noise factor (see Table 5.1). Cost is

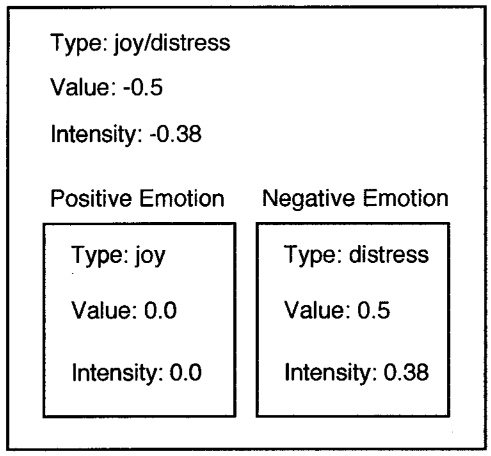

Figure 5.3: Emotion instance data structure for joy/distress

decreased because the agent becomes more willing to take risks just to achieve the goal, and noise is increased to simulate the effects of stress [SJW+02].

5.5 Emotional State

The emotional state of an E-JAM agent is represented by a number of emotion instances. An agent has one instance each of global emotion dimensions and one instance for each goal of local emotion dimensions. In practice this means that an agent has one instance each of joy/distress, satisfaction/disappointment, relief/fears-confirmed, and frustration. For each goal, there is an associated hope/fear emotion instance.

Figure 5.3 shows an emotion instance data structure for joy/distress. Each emotion instance contains a *value* between -1.0 and 1.0, corresponding to a value between 0.0 and 1.0 for either the positive or negative emotion in the emotion instance. When the value is less than zero, then the negative emotion instance has a value corresponding to the emotion dimension value. Similarly, when the emotion dimension's value is greater than zero, then the positive emotion instance has a value defined by the emotion dimension value. This property ensures that both the positive and negative emotions cannot be active at the same time. For example, Figure 5.3 shows a value of -0.5 for the emotion instance, corresponding to a value of 0.5 for the negative distress emotion. The positive joy emotion must have a value of 0.0. Similarly, each emotion instance has an *intensity* which is calculated from the value and the personality activation thresholds defined for the emotion types in question (see Equation 5.1, below). The distress emotion in the example figure has an intensity of 0.38, so the joy/distress emotion instance has an overall intensity of -0.38. Again, the positive

Intensity Range	Granular Scale	World Model Integer
$[-1, -0.66)$	high negative	-3
$[-0.66, -0.33)$	medium negative	-2
$[-0.33, 0)$	low negative	-1
0	inactive	0
$(0, 0.33]$	low positive	1
$(0.33, 0.66]$	medium positive	2
$(0.66, 1]$	high positive	3

Table 5.3: World Model representation of intensity

joy emotion must have an intensity of 0.0 because it is the opposite of distress.

An emotion instance is *active* if the value of one of its emotions is greater than the activation threshold of its associated emotion type. In other words, an emotion instance is active if it has non-zero intensity. We compute the intensity for an emotion e is by the following equation.

$$e.\,\text{Intensity} = \begin{cases} \frac{e.\,\text{Value} - e.\,\text{Threshold}}{1.0 - e.\,\text{Threshold}} \text{ if } e.\,\text{Value} > e.\,\text{Threshold} \\ 0.0 \text{ otherwise} \end{cases} \tag{5.1}$$

With this equation, intensity for an emotion ranges from 0.0 to 1.0, no matter what the activation threshold. When a valenced component of an emotion instance has non-zero intensity, the emotion instance has intensity of the same magnitude and valence. For example, if joy (the positive component) has intensity 0.5, then joy/distress has intensity 0.5. If distress (the negative component) has intensity 0.4, then joy/distress has intensity -0.4. The intensity for an emotion instance ranges from -1.0 to 1.0. As we shall see in Section 5.8, only active emotions can impact the utility computations or trigger E-Plans.

In E-JAM plans, it is useful to have a more coarse-grained scale for emotion instance intensities than floating point numbers from 0.0 to 1.0. We represent emotion instance intensities on a valenced granular scale of "inactive," "low," "medium," and "high." We write an integer form of the scale to the E-JAM world model as a representation of the emotional state of the agent. The integer form of the scale provides ease of comparison in E-JAM plans and conditionals. Table 5.3 shows how emotion instance intensities are converted to E-JAM world model entries. For example, the emotion instance shown in Figure 5.3 would have a world model entry of -2, representing "medium negative," because it has an intensity of -0.38.

```
PLAN: {
NAME: "use_toothbrush"
GOAL: ACHIEVE clean_floor;
ATTRIBUTES: "Type C Cost 0.5 Prob 0.8"
PRECONDITION:
        FACT tried "use_toothbrush" 0;
BODY:
        EXECUTE println "He could not see a toothbrush nearby.";
        EXECUTE recordSubgoal;
        ACHIEVE have "toothbrush" :UTILITY 0.5;
        EXECUTE println "He scrubbed the floor with the toothbrush.";
EFFECTS:
        UPDATE (tried "use_toothbrush" 0) (tried "use_toothbrush" 1);
        EXECUTE println "Finally the floor was clean and shiny.";
FAILURE:
        UPDATE (tried "use_toothbrush" 0) (tried "use_toothbrush" 1);
        EXECUTE println "Without a toothbrush, he failed his task.";
}
```

Figure 5.4: Example C-Plan to clean the floor

5.6 Goals and Plans

Before we discuss how emotional state is generated and how it influences agent behaviour by affecting utility and triggering E-Plans, we must provide some details about how goals and plans are represented in E-JAM. The researcher defines the external achievement or maintenance goals for the agent, along with initial world state and a set of plans, in a plan library file. In E-JAM, goals and plans have additional attributes and restrictions that are not present in JAM. Any plan can have precondition or context requirements, *including conditions on emotional state*. As mentioned previously, a representation of emotional state is posted to the World Model for implementation simplicity.

Figure 5.4 shows a simple example C-Plan for an agent to achieve the goal of cleaning the floor by using a toothbrush. The type of plan is given in the attributes section. The plan is associated with a World Model entry `tried` which the precondition section checks to ensure that the plan is only attempted once. When the plan succeeds or fails, the World Model entry is updated to indicate that the plan has been attempted. The C-Plan has a single sub-goal, where the agent must acquire a toothbrush using some other C-Plan.

Goals assigned to the agent have a direct utility defined in JAM, which is interpreted by E-JAM as the *value* of the goal. The final utility computation requires other variables which are defined on each plan. All plans have pre-set utility attributes for cost and probability of success. These utility attributes are subject to modification during the episode as a function of emotional state. Cost and probability of success are specified with attribute values between 0.0 and 1.0. The plan in Figure 5.4 has "Cost 0.5 Prob 0.8," indicating that

```
PLAN: {
NAME: "dance"
GOAL: ACHIEVE InternalState "homeostasis" "satisfied";
ATTRIBUTES: "Type E Prob 0.5 Cost 0.2
  onsuccess joy -0.4 frustration -0.1
  onfail    frustration 0.2"
PRECONDITION:
  FACT InternalState "joy/distress" $joyintensity;
  (> $joyintensity 1);
BODY:
  EXECUTE println "He began to dance happily.";
  EXECUTE random $fail;
  TEST (< $fail 0.5);
EFFECTS:
  EXECUTE println "After a few minutes the jig was up.";
FAILURE:
  EXECUTE println "He lost his footing and fell down."
}
```

Figure 5.5: Example E-Plan to dance for joy

it involves a medium cost and a high probability of success.

E-JAM agents make use of two kinds of plans: C-Plans and E-Plans. C-Plans are cognitive plans that primarily affect the external world. They can have any kind of goal and can become applicable by the usual JAM rules. E-Plans are emotional plans that are modelled on the principle of homeostasis. E-Plans are activated when the agent's emotional state is active, and primarily act to reduce the emotional state. The type of plan is specified with an attribute pair "Type C" for C-Plans or "Type E" for E-Plans. Figure 5.5 shows an example E-Plan for dancing when the agent is joyful. The E-Plan has a precondition that the intensity of joy/distress must be greater than 1 (a World Model value representing "low positive" intensity). This particular E-Plan simply succeeds or fails randomly. In general, E-Plans can be as complicated or as simple as the scenario designer wishes.

In E-JAM we have implemented E-Plans as regular JAM plans with a fixed goal and extended attributes. Every agent that supports E-Plans must define a special top level goal for maintaining homeostasis. This goal is represented in the world model by the statement:

```
MAINTAIN InternalState "homeostasis" "satisfied";
```

The world model relation `InternalState "homeostasis"` is updated on every cycle by the emotion module, and becomes "unsatisfied" whenever *any* global emotion instance is active. At that point, the `MAINTAIN` goal becomes a concern to the agent, and it must intend a plan to achieve it. Essentially, it does something to dissipate or express the emotion so that a "neutral" state is maintained. The only applicable plans for the goal are E-Plans, which must have the following goal:

```
ACHIEVE InternalState "homeostasis" "satisfied"
```

E-Plans whose preconditions are satisfied become available for the agent to intend and execute in order to achieve the `MAINTAIN` goal. The goal does not have any fixed value because the value of individual E-Plans is computed instead, as we discuss in Section 5.8. The value of an E-Plan depends on the emotional state of the agent and the success effects of the plan.

E-Plans have additional attributes for emotional effects that determine how the agent's emotional state will be impacted by the plan's success or failure. For example, the E-Plan in Figure 5.5 has an *onsuccess* attribute that indicates it decreases joy and frustration when it succeeds. The E-Plan also has an *onfail* attribute that indicates it increases frustration when it fails. The success effects of an E-Plan are also used to compute the value of an E-Plan for the purpose of determining its utility. We discuss these elements in more detail in the following sections. E-Plans, like any other plans, may have preconditions and/or context requirements. We typically use preconditions with E-Plans to target them to particular emotional states. The E-Plan in Figure 5.5, for example, requires that the agent have at least a medium positive intensity for the joy/distress emotion instance.

5.7 Emotion Appraisal

Emotion appraisal is the process where the emotional state of the agent is updated in response to events or other changes. In E-JAM, emotion appraisal is divided into two parts. The Well-being, Frustration, and Prospect-based Confirmation and Disconfirmation emotion instances in our model are updated only when events occur in the world. These emotions are updated in the *event appraisal* process, which runs on any cycle where an event occurs. The Prospect-based hope/fear emotion instances, however, are not updated directly in response to events. Hope and fear depend on the probability of success of goals, which we re-compute in the *prospect appraisal* process from the probability of success of available plans. When the probability of success for a goal changes significantly, we update the hope/fear emotion instance associated with that goal. The prospect appraisal process runs on every E-JAM cycle.

5.7.1 Prospect Appraisal

The hope/fear emotion dimensions are directly influenced by the probability of success of associated goals. In order to appraise hope/fear, we therefore need a way to compute the probability of success of a goal. In E-JAM there is no direct number we can reference for this probability. However, the probability of success of each plan is specified in its attributes. By considering the available plans for a goal, we can compute an estimate for its probability of success, given a few assumptions.

The first assumption we make for computing probability of success for a goal is that each plan may only be attempted once. If plans may be attempted an unlimited number of times, it is easy to show that the probability of success for the goal is always 1.0. In our scenarios we define single-attempt plans so that agents do not behave in a repetitive manner. We also make the assumption that plans are independent from one another. It is easy to break this assumption in a scenario, for example by making all plans involve the same critical step. However, it does not require unreasonable effort to avoid a great deal of dependence when designing a scenario.

Algorithm 1 Prospect Appraisal

Require: List of goals
Require: Applicable Plan Lists for each goal: $p_0, p_1, ..., p_{N-1}$
Ensure: All hope/fear emotion instances are updated based on status of goals, previous emotional state, and available plans.

for $g \in$ Goals **do**
  $P_g \Leftarrow 1 - \prod_{i=0}^{N-1} (1 - P(p_i))$
  $e \Leftarrow$ hope/fear emotion instance associated with g
  if $g.\mathrm{LastP} \neq P_g$ **then**
    $e.\mathrm{Value} \Leftarrow e.\mathrm{Value} + (P_g - g.\mathrm{LastP}) \cdot (1.0 + g.\mathrm{Value})$
    $g.\mathrm{LastP} \Leftarrow P_g$
  end if
end for

The estimated probability of success for all goals is updated on every cycle. If a change occurs in the probability of success for a goal, the associated hope/fear emotion instance is updated. Algorithm 1 specifies the computation of goal probability of success, as well as the emotion value update computation. To compute the probability of success for a goal, P_g, we simply compute the probability that all the available plans for g will fail, and take its complement. We then check to see if P_g is different from the computation of P_g in the previous cycle (LastP). Initially the LastP variable is set to 0.5 for all goals. We update the value of the goal's associated hope/fear emotion instance using the change in probability of success and the goal value. The result of the algorithm is that the probability of success for each goal is up to date, and the values of hope/fear emotion instances reflect any changes since the last E-JAM cycle.

Symbol	Variable
G	Goal Value
C	Plan Cost
P	Plan Probability of success
π	Intensity of prerequisite emotion instance

Symbol	Personality Factor
f_s	Success Factor
f_{gf}	Goal Failure Factor
f_{pf}	Plan Failure Factor
f_G	Goal Value Factor
f_C	Cost Factor
f_P	Probability Factor
f_π	Prerequisite Factor

Table 5.4: Symbols used in Event Appraisal

5.7.2 Event Appraisal

All of the other emotion instances in E-JAM are updated in response to events in the world. The amount of change in the value of an emotion instance is determined by different factors depending on agent personality, the type of event, whether we are evaluating an E-Plan or C-Plan, and whether the emotion instance has a hope/fear prerequisite, anti-requisite, or neither.

Algorithm 2 Event Appraisal (C-Plans)

Require: Goal g related to event of success, goal failure, or plan failure
Require: List of global emotion instances
Ensure: Global emotion instances are updated based on status of goals, agent personality, and previous emotional state.

```
for e ∈ global emotion instances do
  e.Value ⇐ e.Value + EventEffect(g, e)
  if e.Value < −1.0 then
    e.Value = −1.0
  else if e.Value > 1.0 then
    e.Value = 1.0
  end if
end for
```

Events relating to C-Plans are appraised differently in E-JAM from events relating to E-Plans. We first discuss the appraisal process for C-Plans. Algorithm 2 specifies the event appraisal process for C-Plans. When an event occurs, the global emotion instances are updated according to the effect of the event on each individual emotion instance. We limit the range of emotion instance values to the interval [-1.0, 1.0]. For readability, we have separated out the definition of the EventEffect function. EventEffect(g, e) is the amount of change that is made to the value of an emotion instance e in response to an event relating to goal g. The following equations define how EventEffect is computed. Table 5.4 lists the variables and personality factors used in the computations. The symbols correspond to the personality factors defined for each emotion dimension, such as the one in Figure 5.2.

EventEffect(g, e) is computed by taking the product of $E(g, e)$ and the personality factor that is relevant to the event. $E(g, e)$ represents the emotional effect of the event on the emotion instance e. It consists of the normalized sum of each intensity variable associated with the plan/goal multiplied with the respective personality factor. For instance, the goal value G is multiplied with the personality factor f_G. In this way we capture the linear effect of utility variables given a personality setting. Suppose the personality factor f_G is set to 1.0. Then a larger goal value results in a larger effect on the emotion. Likewise, when f_G is set to -1.0, a larger goal value results in a smaller effect on the emotion. The combination of all the variables in this way gives a representation of the emotional effect of the event. With the following equations for EventEffect(g, e) and $E(g, e)$ we can support the set of emotion dimensions that we use in our model.

$$\text{EventEffect}(g, e) = \begin{cases} f_s \cdot E(g, e) \text{ if the goal succeeded} \\ f_{gf} \cdot E(g, e) \text{ if the goal failed} \\ f_{pf} \cdot E(g, e) \text{ if the plan failed} \end{cases} \tag{5.2}$$

We compute $E(g, e)$ differently depending on whether the emotion instance has a prerequisite or anti-requisite for hope/fear, and whether the hope/fear emotion instance associated with the event's goal is active or not. If an emotion instance has a hope/fear anti-requisite and hope/fear is active, or the emotion instance has a hope/fear prerequisite and hope/fear is inactive, then $E(g, e) = 0$ because the conditions for updating the emotion instance have not been met.

If the emotion instance has neither a hope/fear prerequisite nor anti-requisite, then hope/fear does not influence the computation of $E(g, e)$. In this case, we compute $E(g, e)$ by the following equation:

$$E(g, e) = \frac{(f_G \cdot G) + (f_C \cdot C) + (f_P \cdot P)}{|f_G| + |f_C| + |f_P|} \tag{5.3}$$

If the emotion instance has a prerequisite for hope/fear and hope/fear is active, or if it has an anti-requisite for hope/fear and hope/fear is inactive, then we factor in the intensity of the goal's associated hope/fear when computing $E(g, e)$. In this case, we compute $E(g, e)$

by the following equation:

$$E(g,e) = \frac{(f_G \cdot G) + (f_C \cdot C) + (f_P \cdot P) + (f_\pi \cdot \pi)}{|f_G| + |f_C| + |f_P| + |f_\pi|} \tag{5.4}$$

Let us illustrate the above equations with an appraisal of the example C-Plan given in Figure 5.4, which has $C = 0.5$ and $P = 0.8$. Suppose that the goal has value 0.4, so that $G = 0.4$. Our agent has the emotion dimension for joy/distress given in Figure 5.2, which has $f_G = 1.0$, $f_C = 0.0$, and $f_P = 1.0$. Further suppose the agent does not have active hope/fear for the goal. We can compute $E(g, e)$ using Equation 5.3 as follows.

$$E(g,e) = \frac{(1.0 \cdot 0.4) + (0.0 \cdot 0.5) + (1.0 \cdot 0.8)}{|1.0| + |0.0| + |1.0|} = \frac{1.2}{2.0} = 0.6$$

Then we can compute what EventEffect(g, e) would be for the cases of success, goal failure, and plan failure, using Equation 5.2 as follows.

$$\text{EventEffect}(g,e) = \begin{cases} 1.0 \cdot 0.6 = 0.6 \text{ if the goal succeeded} \\ -1.0 \cdot 0.6 = -0.6 \text{ if the goal failed} \\ 0.0 \cdot 0.6 = 0.0 \text{ if the plan failed} \end{cases}$$

EventEffect(g, e) is then used to update the value of the joy/distress emotion instance, as in Algorithm 2. Note that since f_{pf} is 0.0 for joy/distress (see Figure 5.2), plan failure will never affect the value of joy/distress no matter how large $E(g, e)$ is. The emotion dimension is defined in this way because it was based on the emotion types from the OCC model. However, the frustration emotion dimension is one that is affected by plan failure.

The event appraisal for E-Plans is very straightforward. Each E-Plan has attributes that specify the direct effects of the E-Plan when it succeeds or fails. These effects consist of changes in the value of any number of global emotion instances. Instead of directly affecting the value of emotion instances, the effects of E-Plans affect the value of the valenced components of emotion instances. For example, the E-Plan in Figure 5.5 decreases the value of joy by 0.4 when it succeeds. If the agent has a positive value for joy/distress, the value will be decreased by 0.4 (to a minimum of 0.0). However, if the value of joy/distress is 0.0 or lower, it will remain unchanged. The same E-Plan decreases the value of frustration by 0.1 when it succeeds, and increases the value of frustration by 0.2 when it fails. We constrain the sum of effects for a particular E-Plan by the following:

$$\left| \sum_{\text{emotions } e} \text{effect}_s(e)) \right| \leq 1 \tag{5.5}$$

$$\left| \sum_{\text{emotions } e} \text{effect}_f(e)) \right| \leq 1 \tag{5.6}$$

where $\text{effect}_s(e)$ is the effect on emotion e when the E-Plan succeeds and $\text{effect}_f(e)$ is the effect on emotion e when the E-Plan fails. We use these constraints in order to ensure that utility computations for E-Plans remain comparable to C-Plans.

The result of the appraisal process, including both prospect and event appraisal, is that the agent's emotional state is updated to reflect the agent's feelings about what is happening in the world. The agent's personality, plan library, current goals, and current emotional state all influence the appraisal computations. The appraisal process is the only place in E-JAM where the agent's emotional state is changed.

5.8 Emotional Impact on Decision-Making

E-JAM provides two ways for emotions to influence the behaviour of an agent. In Section 5.6, we described E-Plans and how they are supported in E-JAM. This section gives details about how an agent's emotional state has an impact on its cognitive decision-making. A general mechanism for decision-making is already present in JAM. For each current goal that has no intention, applicable plans are added to the APL (Applicable Plan List). The utility of each APL element is calculated and the plan with the highest utility is intended. If multiple intentions are available for execution, their utilities are calculated and the one with the highest utility is executed.

Normally, JAM computes utility as the sum of goal utility and plan utility. E-JAM uses a utility computation based on ACT-R [Bel01] that takes emotional state into account. We compute utility slightly differently for C-Plans and E-Plans. The utility equation we use is the same for each type of plan, but the way we determine the utility variables is different. The utility variables used for C-Plans are primarily based on variables external to the agent (i.e. from the world), but are also influenced by the agent's emotional state. The variables we use for E-Plans are based to a greater extent on the agent's internal emotional state. It is worth noting here, however, that E-Plans and C-Plans compete on equal footing as far as E-JAM is concerned. Each plan, no matter the type, has a final computed utility which is directly compared to the utility of other plans.

The ACT-R expected utility E for a plan is computed as follows:

$$E = P \cdot G - C + \xi(\tau) \tag{5.7}$$

where P is the expected probability of achieving the goal using this plan, G is the value of the current goal, C is the cost of executing this plan, and $\xi(\tau)$ is a random variable representing noise.

The values of the variables P and C are computed starting with the pre-set probability of success and cost attributes of the plan in question. $\xi(\tau)$ is first randomly generated using the agent's noise personality parameters, and G has an initial setting of the related goal's value. We apply a combination of the agent's personality and emotional state to determine the final values of these utility variables. The final variable values are used to compute the utility for the plan. In the case of E-Plans, G is determined in a different way (discussed shortly).

For emotional influence on utility variables, we use a modified winner-takes-all approach where only the largest impact on each variable is used. Thus, it is not necessarily the most intense emotion that is used, but the emotion that has the largest impact given intensity and personality parameters. This method allows a number of low intensity emotions to combine to result in a strong reaction, since more than one emotion can impact the utility computation. The emotion instances considered for utility computations are all *active* global emotion instances and the *active* local emotion instances associated with the plan's goal. For example, we might calculate that the perceived cost of a plan would be increased by 0.3 if frustration is used as the emotional impact. Suppose that the perceived cost of the plan would be decreased by 0.4 if relief/fears-confirmed is used as the emotional impact. In this case, we would compute the perceived cost using the impact of relief/fears-confirmed because it causes the largest change in the perception of cost. The emotion instances that impact the other utility variables need not be the same as the one that impacts cost.

We compute the final variables P, G, C, and $\xi(\tau)$ by the following equations:

$$P = p + \max_{\text{emotions } e} p \cdot e.\text{Intensity} \cdot e.\text{ProbabilityImpact} \tag{5.8}$$

$$G = g + \max_{\text{emotions } e} g \cdot e.\text{Intensity} \cdot e.\text{GoalValueImpact} \tag{5.9}$$

$$C = c + \max_{\text{emotions } e} c \cdot e.\text{Intensity} \cdot e.\text{CostImpact} \tag{5.10}$$

$$\xi(\tau) = n + \max_{\text{emotions } e} n \cdot e.\text{Intensity} \cdot e.\text{NoiseImpact} \tag{5.11}$$

where p is the initial expected probability of success, g is the initial goal value, c is the initial cost, and n is the initial noise value. P, G, and C are each then constrained to the range $[0.0, 1.0]$. $\xi(\tau)$ is limited to the range $[-1.0, 1.0]$.

As mentioned earlier, the utility variable G is computed differently for E-Plans than for C-Plans. The reason is that E-Plans all have the same goal, and thus goal value is not a relevant distinguishing feature between E-Plans (of which there may be several). Furthermore, since E-Plans are modelled after the process of homeostasis, the value of an E-Plan to an agent corresponds to the effectiveness of the E-Plan in returning the agent to the neutral emotional state. Therefore we compute the value of an E-Plan by determining its effect on the agent's emotional state when it succeeds. Again we use a winner-takes-all method by computing the largest reduction in emotion value, modulated by the intensity of the affected emotion. For E-Plans, G is computed by the following equation:

$$G = \max_{\text{emotions } e} (-\,\text{effect}_s(e)) \cdot (1.0 + e.\,\text{Intensity}) \tag{5.12}$$

The emotionally-influenced utility computed by the equations above is used in E-JAM to replace the utility in the usual JAM decision-making algorithm. In this way we extend the decision-making algorithm to support emotional influence without changing the fundamentals of how it works. C-Plans and E-Plans both compete on equal footing, since they have a similar utility computation and the JAM algorithm does not make any distinction between them.

5.9 Example E-JAM Trace

Many of the concepts introduced in E-JAM can be illustrated with an example trace output from the system. The annotated trace below was created using a scenario with the same C-Plan structure as in the JAM example trace in Section 4.2.7. In this example we also include several simple E-Plans. The trace is produced by an E-JAM agent with a "neutral" personality, which means all of its emotion types have the same activation threshold of 0.3. We use our basic personality setup for the other personality parameters and set the noise parameters to zero. Elements of the trace specific to E-JAM include emotionally-influenced utility computation, prospect and event appraisal, emotional state, and E-Plan selection.

Figure 5.6 shows a graphical representation of the behaviour of the agent. The format of the diagram is the same as in the JAM example, except that E-Plans are now included. Dashed arrows indicate where the agent decides to execute an E-Plan instead of a C-Plan. The diagram is admittedly more complicated than in the JAM trace, but that is because the agent's behaviour is more complicated. For example, consider what happens to the agent during "Subgoal 1." We can see that the agent starts to work on "Plan 1" on cycle 10, and the plan fails on cycle 15. Next, on cycle 16, the homeostasis goal becomes active and an E-Plan for taking a walk is selected. The E-Plan succeeds on cycle 19, removing the

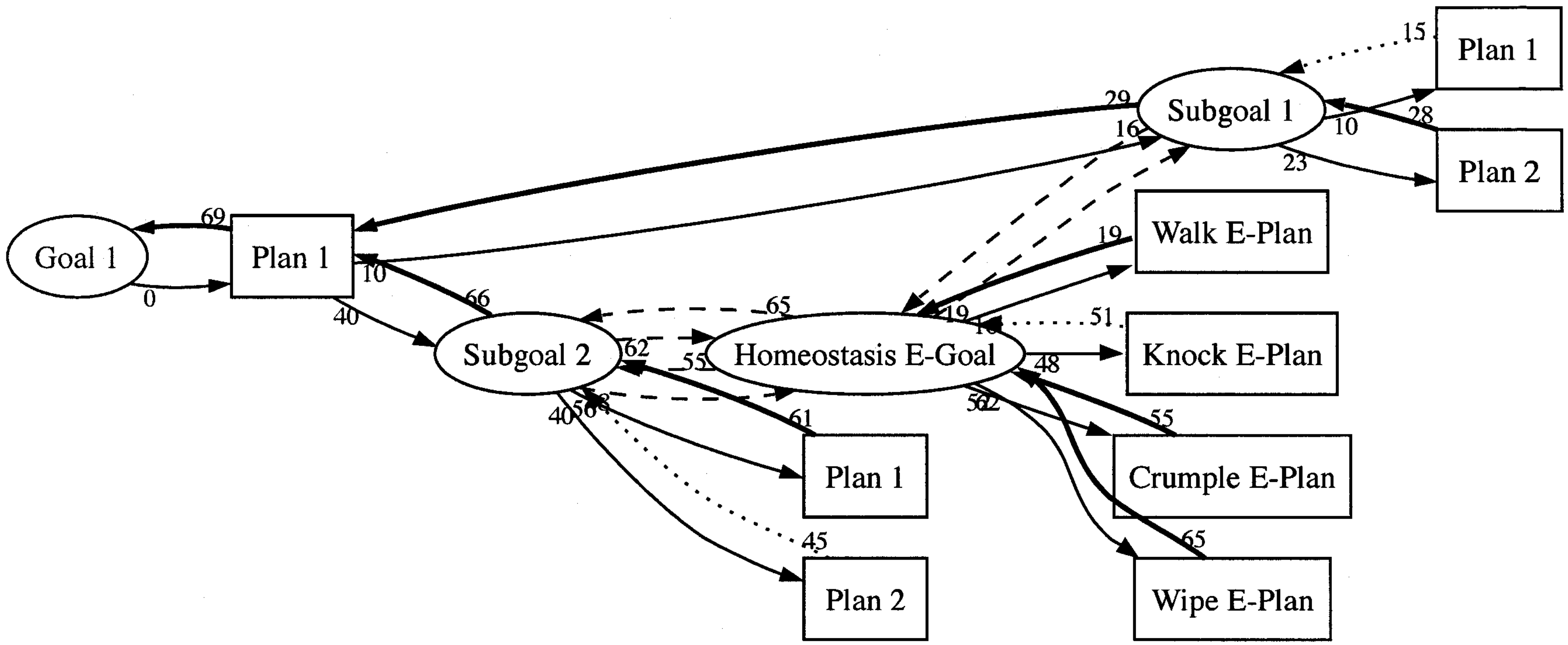

Figure 5.6: Graphical representation of agent behaviour in E-JAM. Ellipses represent goals and boxes represent plans. Edges from left to right indicate sub-goaling or plan selection. Bold edges and dotted edges from right to left indicate success and failure, respectively. Dashed edges indicate where the agent decides to execute an E-Plan instead of a C-Plan.

frustration experienced by the agent and returning the agent to the neutral state. The agent then returns to "Subgoal 1" to try "Plan 2."

Some parts of the trace output may need clarification. In appraisal steps, we see how much the values of emotion instances change. This may or may not cause the emotion instances to become active, because each emotion type has an activation threshold of 0.3 for this agent. When a plan fails, we see, for example, "frustration decreased by 0.35." This means that the value of the *emotion instance* for frustration is decreased. Since the only valenced component of the frustration emotion instance is negative frustration, this means that the agent is *more* frustrated when the value of the emotion instance is decreased. On each cycle, the emotional state of the agent is reported by listing the intensity of all active emotion instances. The magnitude and valence of the intensity is shown by the world model granular scale value and component of the emotion instance, respectively. For local hope/fear emotion instances, we also note the goal associated with the emotion instance. All unlisted emotion instances have zero intensity (but not necessarily zero value).

```
1  JAM Parser Version 65 + 76i:
2  JAM definition parse successful.
3
4  Interpreter: starting cycle 0
5  Active emotions:
6  Interpreter: Deciding on a plan
7  Emotion Module: Appraising Prospects
8    'goal1' hope/fear increased by  0.63
9  Computing utility for plan 'Goal 1 Plan 3'
10   Original values: G =  0.50, P =  0.80, C =  0.80, N =  0.00
11   G winner is hope/fear, G changed by  0.24
12   Final values:    G =  0.74, P =  0.80, C =  0.80, N =  0.00
13   Final utility = -0.21
14 Computing utility for plan 'Goal 1 Plan 2'
15   Original values: G =  0.50, P =  0.50, C =  0.50, N =  0.00
16   G winner is hope/fear, G changed by  0.24
17   Final values:    G =  0.74, P =  0.50, C =  0.50, N =  0.00
18   Final utility = -0.13
19 Computing utility for plan 'Goal 1 Plan 1'
20   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
21   G winner is hope/fear, G changed by  0.24
22   Final values:    G =  0.74, P =  0.20, C =  0.20, N =  0.00
23   Final utility = -0.05
24 Interpreter: Selected plan "Goal 1 Plan 1" from APL.
25 Interpreter: Executing the intention structure.
26 IntentionStructure: Executing goal goal1
27 AGENT: Goal 1 Plan 1
28
29 Interpreter: starting cycle 1
30 Active emotions:
31   hope (goal1): medium
32 Interpreter: Deciding on a plan
33 Interpreter: Executing something already in the intention structure.
34 IntentionStructure: Executing goal goal1
```

After loading the Plan Library, the Interpreter has a single goal for the agent, "Goal 1." There are three applicable plans for the goal. First, prospect appraisal occurs and we see that the agent is hopeful about achieving the goal (line 8). The rest of the agent's emotional state is initially neutral, so the utility computations are only affected by hope/fear. We see that hope/fear increases goal value and has no effect on the other variables. The final utility is computed using the final utility variables. "Plan 1" is selected because it has the highest final utility (line 24). Once the plan is selected, the Intention Structure starts to execute the plan.

```
35 Interpreter: starting cycle 10
36 Active emotions:
37   hope (goal1): medium
38 Interpreter: Deciding on a plan
39 Emotion Module: Appraising Prospects
40   'goal1plan1subgoal1' hope/fear decreased by  0.45
41 Computing utility for plan 'Goal 1 Plan 1 Subgoal 1 Plan 2'
42   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
43   G winner is hope/fear, G changed by -0.11
44   Final values:    G =  0.39, P =  0.20, C =  0.20, N =  0.00
45   Final utility = -0.12
46 Computing utility for plan 'Goal 1 Plan 1 Subgoal 1 Plan 1'
47   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
48   G winner is hope/fear, G changed by -0.11
49   Final values:    G =  0.39, P =  0.20, C =  0.20, N =  0.00
50   Final utility = -0.12
```

```
51 Interpreter: Selected plan "Goal 1 Plan 1 Subgoal 1 Plan 1" from APL.
52 Interpreter: Executing the intention structure.
53 IntentionStructure: Executing goal goal1plan1subgoal1
54 AGENT: Goal 1 Plan 1 Subgoal 1 Plan 1
55
56 Interpreter: starting cycle 11
57 Active emotions:
58   hope (goal1): medium
59   fear (goal1plan1subgoal1): low
60 Interpreter: Deciding on a plan
61 Interpreter: Executing something already in the intention structure.
62 IntentionStructure: Executing goal goal1plan1subgoal1
```

After executing "Plan 1" for several cycles, the plan sub-goals to "Subgoal 1." As before, the agent appraises the prospects for the goal. This time, the agent is fearful that the sub-goal will fail (line 40) because the available plans have a low probability of success. The utility for each plan is computed including the influence of hope/fear. The Interpreter then decides between the applicable plans for the goal. Since both applicable plans have the same utility, -0.12, the agent randomly selects between them, choosing "Plan 1" (line 51). The intended plan is then executed by the Intention Structure.

```
63 Interpreter: starting cycle 15
64 Active emotions:
65   hope (goal1): medium
66   fear (goal1plan1subgoal1): low
67 Interpreter: Deciding on a plan
68 Interpreter: Executing something already in the intention structure.
69 IntentionStructure: Executing goal goal1plan1subgoal1
70 Appraising Event: Plan failure: Goal 1 Plan 1 Subgoal 1 Plan 1
71   frustration decreased by  0.35
```

After a few cycles, the plan for "Subgoal 1" fails (line 70). The plan failure event is appraised and results in the agent becoming more frustrated (line 71).

```
72 Interpreter: starting cycle 16
73 Active emotions:
74   frustration: low
75   hope (goal1): medium
76   fear (goal1plan1subgoal1): low
77 Interpreter: Deciding on a plan
78 Emotion Module: Appraising Prospects
79 Computing utility for plan 'E-Plan Knock a box over'
80   Original values: G =  0.00, P =  0.50, C =  0.20, N =  0.00
81   E-plan goal value winner is frustration, new G =  0.21
82   C winner is frustration, C changed by -0.01
83   Final values:    G =  0.21, P =  0.50, C =  0.19, N =  0.00
84   Final utility = -0.08
85 Computing utility for plan 'E-Plan Crumple up papers'
86   Original values: G =  0.00, P =  0.80, C =  0.50, N =  0.00
87   E-plan goal value winner is frustration, new G =  0.32
88   C winner is frustration, C changed by -0.04
89   Final values:    G =  0.32, P =  0.80, C =  0.46, N =  0.00
90   Final utility = -0.21
91 Computing utility for plan 'E-Plan Go for a walk'
92   Original values: G =  0.00, P =  0.50, C =  0.20, N =  0.00
93   E-plan goal value winner is frustration, new G =  0.32
94   C winner is frustration, C changed by -0.01
95   Final values:    G =  0.32, P =  0.50, C =  0.19, N =  0.00
96   Final utility = -0.03
97 Interpreter: Selected plan "E-Plan Go for a walk" from APL.
```

```
98  Computing utility for plan 'Goal 1 Plan 1'
99    Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
100   G winner is hope/fear, G changed by  0.24
101   C winner is frustration, C changed by -0.01
102   Final values:    G =  0.74, P =  0.20, C =  0.19, N =  0.00
103   Final utility = -0.04
104 Interpreter: Executing the intention structure.
105 IntentionStructure: Executing goal InternalState
106 Agent: Starting E-Plan Go for a walk
```

On cycle 16, the agent has an active global emotional state (a low level of frustration), so the homeostasis E-Goal comes into play. The agent must first decide which applicable E-Plan to intend to the goal. After computing the emotionally-influenced utility for each E-Plan, it decides to use the plan to go for a walk, which has final utility -0.03. The agent now has two intention threads – the original thread for "Goal 1" and the new thread for the E-Goal. It re-computes the utility for the leaf intention on the "Goal 1" thread, which is now "Plan 1" (lines 98–103), and finds that it is slightly lower than the utility of the intended E-Plan. Therefore, the agent executes the E-Plan, indicated by the goal named "InternalState" (lines 105–106).

```
107 Interpreter: starting cycle 19
108 Active emotions:
109   frustration: low
110   hope (goal1): medium
111   fear (goal1plan1subgoal1): low
112 Interpreter: Deciding on a plan
113 Interpreter: Executing something already in the intention structure.
114 IntentionStructure: Executing goal InternalState
115 Appraising Event: Goal success: InternalState
116   frustration increased by  0.30
117
118 Interpreter: starting cycle 20
119 Active emotions:
120   hope (goal1): medium
121   fear (goal1plan1subgoal1): low
122 Interpreter: Deciding on a plan
123 Interpreter: Executing something already in the intention structure.
124 IntentionStructure: Executing goal goal1
```

On line 115, we see that the E-Goal succeeds and results in the agent's frustration being partially dissipated (recall its value was -0.35, and now it is increased by 0.30 for a final value of -0.05). This E-Plan would also dissipate other negative emotions by 0.20, but they all have zero value at this point so they remain unchanged. We can see on the next cycle that the agent's emotional state is now back to neutral, because the frustration emotion instance is inactive (its value is 0.05 versus a threshold of 0.30). Therefore the agent returns to its remaining intention thread for "Goal 1" (line 124).

```
125 Interpreter: starting cycle 23
126 Active emotions:
127   hope (goal1): medium
128   fear (goal1plan1subgoal1): low
129 Interpreter: Deciding on a plan
130 Emotion Module: Appraising Prospects
```

```
131 Computing utility for plan 'Goal 1 Plan 1 Subgoal 1 Plan 2'
132   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
133   G winner is hope/fear, G changed by -0.11
134   Final values:    G =  0.39, P =  0.20, C =  0.20, N =  0.00
135   Final utility = -0.12
136 Interpreter: Selected plan "Goal 1 Plan 1 Subgoal 1 Plan 2" from APL.
137 Interpreter: Executing the intention structure.
138 IntentionStructure: Executing goal goal1plan1subgoal1
139 AGENT: Goal 1 Plan 1 Subgoal 1 Plan 2
```

Now the agent tries to achieve "Subgoal 1" again. This time, the only applicable plan is "Plan 2" because it is only permitted to try each plan once. It selects the plan and begins to execute it (line 139).

```
140 Interpreter: starting cycle 28
141 Active emotions:
142   hope (goal1): medium
143   fear (goal1plan1subgoal1): low
144 Interpreter: Deciding on a plan
145 Interpreter: Executing something already in the intention structure.
146 IntentionStructure: Executing goal goal1plan1subgoal1
147 Appraising Event: Goal success: goal1plan1subgoal1
148   relief/fears-confirmed increased by  0.21
149   frustration increased by  0.35
150
151 Interpreter: starting cycle 29
152 Active emotions:
153   hope (goal1): medium
154 Interpreter: Deciding on a plan
155 Interpreter: Executing something already in the intention structure.
156 IntentionStructure: Executing goal goal1
```

On line 147 we see that the "Plan 1 Subgoal 2" succeeds. The event is appraised and results in the increase of relief/fears-confirmed by 0.21, as well as the dissipation of the remaining value of frustration. Since the relief/fears-confirmed value is below threshold, the emotion instance remains inactive. The agent returns to executing "Plan 1" for "Goal 1" (line 156).

```
157 Interpreter: starting cycle 40
158 Active emotions:
159   hope (goal1): medium
160 Interpreter: Deciding on a plan
161 Emotion Module: Appraising Prospects
162   'goal1plan1subgoal2' hope/fear decreased by  0.45
163 Computing utility for plan 'Goal 1 Plan 1 Subgoal 2 Plan 2'
164   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
165   G winner is hope/fear, G changed by -0.11
166   Final values:    G =  0.39, P =  0.20, C =  0.20, N =  0.00
167   Final utility = -0.12
168 Computing utility for plan 'Goal 1 Plan 1 Subgoal 2 Plan 1'
169   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
170   G winner is hope/fear, G changed by -0.11
171   Final values:    G =  0.39, P =  0.20, C =  0.20, N =  0.00
172   Final utility = -0.12
173 Interpreter: Selected plan "Goal 1 Plan 1 Subgoal 2 Plan 2" from APL.
174 Interpreter: Executing the intention structure.
175 IntentionStructure: Executing goal goal1plan1subgoal2
176 AGENT: Goal 1 Plan 1 Subgoal 2 Plan 2
```

The agent eventually comes to another sub-goal, "Subgoal 2." Again, the utility values for each applicable plan for "Subgoal 2" are the same, so one ("Plan 2") is randomly selected and executed (line 176).

```
177 Interpreter: starting cycle 45
178 Active emotions:
179   hope (goal1): medium
180   fear (goal1plan1subgoal2): low
181 Interpreter: Deciding on a plan
182 Interpreter: Executing something already in the intention structure.
183 IntentionStructure: Executing goal goal1plan1subgoal2
184 Appraising Event: Plan failure: Goal 1 Plan 1 Subgoal 2 Plan 2
185   frustration decreased by  0.35
```

Similar to what happened in "Subgoal 1," the first plan for "Subgoal 2" fails. The plan failure event is appraised and results in the agent becoming frustrated again (line 185).

```
186 Interpreter: starting cycle 46
187 Active emotions:
188   frustration: low
189   hope (goal1): medium
190   fear (goal1plan1subgoal2): low
191 Interpreter: Deciding on a plan
192 Emotion Module: Appraising Prospects
193 Computing utility for plan 'E-Plan Knock a box over'
194   Original values: G =  0.21, P =  0.50, C =  0.20, N =  0.00
195   E-plan goal value winner is frustration, new G =  0.21
196   C winner is frustration, C changed by -0.01
197   Final values:    G =  0.21, P =  0.50, C =  0.19, N =  0.00
198   Final utility = -0.08
199 Computing utility for plan 'E-Plan Crumple up papers'
200   Original values: G =  0.32, P =  0.80, C =  0.50, N =  0.00
201   E-plan goal value winner is frustration, new G =  0.32
202   C winner is frustration, C changed by -0.04
203   Final values:    G =  0.32, P =  0.80, C =  0.46, N =  0.00
204   Final utility = -0.21
205 Interpreter: Selected plan "E-Plan Knock a box over" from APL.
206 Computing utility for plan 'Goal 1 Plan 1'
207   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
208   G winner is hope/fear, G changed by  0.24
209   Final values:    G =  0.74, P =  0.20, C =  0.20, N =  0.00
210   Final utility = -0.05
211 Interpreter: Executing the intention structure.
212 IntentionStructure: Executing goal goal1
```

We see the agent has an active emotional state (frustration, see line 188), so the homeostasis goal is activated and E-Plans are considered for it. Since the E-Plan to go for a walk was already used in the episode, it is not considered again. The one with the highest final utility is knocking over a box, so the agent intends that E-Plan (line 205). However, the final utility for "Plan 1" (-0.05) is higher than the utility for the E-Plan (-0.08), so the agent actually continues to execute the C-Plan instead (line 212).

```
213 Interpreter: starting cycle 48
214 Active emotions:
215   frustration: low
216   hope (goal1): medium
217   fear (goal1plan1subgoal2): low
218 Interpreter: Deciding on a plan
219 Emotion Module: Appraising Prospects
220 Computing utility for plan 'Goal 1 Plan 1 Subgoal 2 Plan 1'
221   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
```

```
222    G winner is hope/fear, G changed by -0.11
223    Final values:     G =  0.39, P =  0.20, C =  0.20, N =  0.00
224    Final utility = -0.12
225  Interpreter: Selected plan "Goal 1 Plan 1 Subgoal 2 Plan 1" from APL.
226  Interpreter: Executing the intention structure.
227  IntentionStructure: Executing goal InternalState
228  AGENT: Starting E-Plan Knock a box over
```

Now the agent tries to achieve "Subgoal 2" again (lines 218-220). This time, the only applicable plan is "Plan 1" because it is only permitted to try each plan once. It intends the plan and then decides which of its intention threads to execute. The utility of the E-Plan is -0.08, while the utility of the C-Plan is -0.12. Effectively, there is higher utility for addressing the low level of frustration than continuing work on the cognitive task. Thus the agent selects the E-Plan and begins to execute it (line 227).

```
229  Interpreter: starting cycle 51
230  Active emotions:
231    frustration: low
232    hope (goal1): medium
233    fear (goal1plan1subgoal2): low
234  Interpreter: Deciding on a plan
235  Interpreter: Executing something already in the intention structure.
236  IntentionStructure: Executing goal InternalState
237  Appraising Event: Plan failure: E-Plan Knock a box over
238    frustration decreased by  0.20
```

Unfortunately for the agent, the E-Plan fails and the agent becomes even more frustrated (lines 237-238). Its E-Goal remains in place because its emotional state is still active.

```
239  Interpreter: starting cycle 52
240  Active emotions:
241    frustration: medium
242    hope (goal1): medium
243    fear (goal1plan1subgoal2): low
244  Interpreter: Deciding on a plan
245  Emotion Module: Appraising Prospects
246  Computing utility for plan 'E-Plan Crumple up papers'
247    Original values: G =  0.32, P =  0.80, C =  0.50, N =  0.00
248    E-plan goal value winner is frustration, new G =  0.41
249    C winner is frustration, C changed by -0.18
250    Final values:     G =  0.41, P =  0.80, C =  0.32, N =  0.00
251    Final utility =  0.00
252  Interpreter: Selected plan "E-Plan Crumple up papers" from APL.
253  Computing utility for plan 'Goal 1 Plan 1 Subgoal 2 Plan 1'
254    Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
255    G winner is hope/fear, G changed by -0.11
256    C winner is frustration, C changed by -0.07
257    Final values:     G =  0.39, P =  0.20, C =  0.13, N =  0.00
258    Final utility = -0.05
259  Interpreter: Executing the intention structure.
260  IntentionStructure: Executing goal InternalState
261  AGENT: Starting E-Plan Crumple up papers
```

The agent considers the remaining applicable E-Plans and computes their utility. Notice that as the agent's frustration grows, its impact on the cost utility variable increases for all plans. The agent intends the E-Plan to crumple up papers and then selects it for execution (lines 252 and 261).

```
262 Interpreter: starting cycle 55
263 Active emotions:
264   frustration: medium
265   hope (goal1): medium
266   fear (goal1plan1subgoal2): low
267 Interpreter: Deciding on a plan
268 Interpreter: Executing something already in the intention structure.
269 IntentionStructure: Executing goal InternalState
270 Appraising Event: Goal success: InternalState
271   frustration increased by  0.30
272
273 Interpreter: starting cycle 56
274 Active emotions:
275   hope (goal1): medium
276   fear (goal1plan1subgoal2): low
277 Interpreter: Deciding on a plan
278 Interpreter: Executing something already in the intention structure.
279 IntentionStructure: Executing goal goal1plan1subgoal2
280 AGENT: Goal 1 Plan 1 Subgoal 2 Plan 1
```

This time, the E-Plan succeeds (line 270) and the agent's frustration is dissipated enough so that it is no longer active (line 271). The agent selects its "Subgoal 2 Plan 1" C-Plan for execution and goes to work on it (line 280).

```
281 Interpreter: starting cycle 61
282 Active emotions:
283   hope (goal1): medium
284   fear (goal1plan1subgoal2): low
285 Interpreter: Deciding on a plan
286 Interpreter: Executing something already in the intention structure.
287 IntentionStructure: Executing goal goal1plan1subgoal2
288 Appraising Event: Goal success: goal1plan1subgoal2
289   relief/fears-confirmed increased by  0.21
290   frustration increased by  0.35
291
292 Interpreter: starting cycle 62
293 Active emotions:
294   relief: low
295   hope (goal1): medium
296 Interpreter: Deciding on a plan
297 Emotion Module: Appraising Prospects
298 Computing utility for plan 'E-Plan Wipe forehead'
299   Original values: G =  0.00, P =  0.80, C =  0.20, N =  0.00
300   E-plan goal value winner is relief/fears-confirmed, new G =  0.23
301   P winner is relief/fears-confirmed, P changed by  0.13
302   Final values:    G =  0.23, P =  0.93, C =  0.20, N =  0.00
303   Final utility =  0.02
304 Computing utility for plan 'E-Plan Sing a song'
305   Original values: G =  0.00, P =  0.50, C =  0.50, N =  0.00
306   E-plan goal value winner is relief/fears-confirmed, new G =  0.47
307   P winner is relief/fears-confirmed, P changed by  0.08
308   Final values:    G =  0.47, P =  0.58, C =  0.50, N =  0.00
309   Final utility = -0.23
310 Computing utility for plan 'E-Plan Do a Dance'
311   Original values: G =  0.00, P =  0.50, C =  0.20, N =  0.00
312   E-plan goal value winner is relief/fears-confirmed, new G =  0.23
313   P winner is relief/fears-confirmed, P changed by  0.08
314   Final values:    G =  0.23, P =  0.58, C =  0.20, N =  0.00
315   Final utility = -0.06
316 Interpreter: Selected plan "E-Plan Wipe forehead" from APL.
317 Computing utility for plan 'Goal 1 Plan 1'
318   Original values: G =  0.50, P =  0.20, C =  0.20, N =  0.00
319   G winner is hope/fear, G changed by  0.24
320   P winner is relief/fears-confirmed, P changed by  0.03
321   Final values:    G =  0.74, P =  0.23, C =  0.20, N =  0.00
```

```
322   Final utility = -0.03
323 Interpreter: Executing the intention structure.
324 IntentionStructure: Executing goal InternalState
325 AGENT: Starting E-Plan Wipe forehead
```

The agent's C-Plan succeeds (line 288), and the appraisal of the event results in an increase in relief/fears-confirmed (line 289) and a further dissipation of frustration (line 290). Now the relief/fears-confirmed emotion instance is active with a low intensity, so again the agent's E-Goal becomes valid. A different set of E-Plans are applicable for positive emotions, and the agent intends and executes an E-Plan for wiping his forehead (lines 316 and 325).

```
326 Interpreter: starting cycle 65
327 Active emotions:
328   relief: low
329   hope (goal1): medium
330 Interpreter: Deciding on a plan
331 Interpreter: Executing something already in the intention structure.
332 IntentionStructure: Executing goal InternalState
333 Appraising Event: Goal success: InternalState
334   relief/fears-confirmed decreased by  0.20
335
336 Interpreter: starting cycle 66
337 Active emotions:
338   hope (goal1): medium
339 Interpreter: Deciding on a plan
340 Interpreter: Executing something already in the intention structure.
341 IntentionStructure: Executing goal goal1
```

On line 333, we see the E-Plan succeeds, partially dissipating the value of the relief/fears-confirmed emotion instance. It is enough to drop the value of the emotion instance below the activation threshold, so the E-Goal is achieved and the agent returns to executing its original C-Plan for "Goal 1" (line 341).

```
342 Interpreter: starting cycle 69
343 Active emotions:
344   hope (goal1): medium
345 Interpreter: Deciding on a plan
346 Interpreter: Executing something already in the intention structure.
347 IntentionStructure: Executing goal goal1
348 Appraising Event: Goal success: goal1
349   satisfaction/disappointment increased by  0.34
350   frustration increased by  0.35
351
352 Interpreter: starting cycle 70
353 Active emotions:
354   satisfaction: low
355 Interpreter: Deciding on a plan
356
357 JAM: All of the agent's top-level goals have been achieved!  Returning...
```

Finally, "Plan 1" for "Goal 1" succeeds (line 348) and the agent accomplishes its initial goal. Since the agent had hope for the goal, it feels satisfied about the event (line 349). However, the agent is finished its work and the simulation ends without any further E-Goals being activated.

5.10 Summary

In this chapter, we presented the E-JAM architecture, which builds on the JAM procedural reasoning framework to add an emotional model component. Our modifications to the JAM code include a new Emotion Module, as well as several changes and improvements throughout the architecture to interface with the Emotion Module. We also give a set of requirements for E-JAM plans that enable support for emotional modelling. E-JAM plans have extended attributes for utility computation and for distinguishing C-Plans and E-Plans. C-Plans are cognitive plans that are used for an agent's problem solving task, while E-Plans are emotional plans that are activated by the agent's internal emotional state.

The E-JAM Emotion Module includes data structures for personality and emotional state, as well as processes for emotion appraisal and for emotionally influenced utility computations. Appraisals take place for prospects of events and for events themselves, and update the agent's emotional state. Utility, which is used for conflict resolution between plans, is influenced by the agent's emotional state. The agent's personality guides the computation in these processes and determines (using thresholds) how much the agent is influenced by its emotional state. When an agent has an active emotional state, it activates an E-Goal (and thus, E-Plans) in order to return itself to a neutral emotional state (the process of homeostasis). When an agent is in a neutral state, it behaves in a more "rational" manner. In the coming chapters we contrast the behaviour of completely rational agents against that of emotional agents who are driven to return to a neutral state.

Chapter 6

Abstract Problem Experiments

6.1 Introduction

Our first experiments evaluate E-JAM using abstractly structured scenarios. This evaluation is intended to confirm the operation of the system and evaluate the patterns of decision-making and changes in emotional state. We develop metrics to characterize the plan selection behaviour of agents. A problem solving scenario in these experiments has three top level goals that are achieved either in a sequential or concurrent mode. We define problem solving experiences in terms of the mix of failures and successes of the three top level goals.

While the abstract problem experiments do not say anything about the believability of agent behaviour, they are useful because they validate the design of the system. We make no hypotheses for the experiments because they aim to provide verification for some of our assumptions. The experiments contrast the behaviour of "rational" and "emotional" agents to verify aspects of personality, emotional state, utility computation, and behaviour selection. We evaluated the abstract scenarios in two ways. We first ran a variety of tests for a large number of episodes to glean statistical information about system operation and agent behaviour. In order to examine more subtle phenomena, we also examined a smaller number of more detailed traces of individual episodes. The detailed analyses explored the effects of emotional state tendencies, E-Plans, and utility computations.

The results of our experiments were mostly as expected. There were four general observations from these experiments:

1. The emotional agents deviate significantly from rational agents both in plan selection and persistence behaviour.

2. A rational agent with a noise factor does not behave similarly to an emotional agent, verifying that emotional modelling produces behaviour distinguishable from noise.

3. An emotional agent that has homeostasis E-Plans behaves differently than agents without them. Specifically, emotional agents with E-Plans make plan selection choices the same as rational agents in many cases.

4. However, the persistence of emotional agents with E-Plans is not consistently higher or lower than the persistence of agents without E-Plans. This keeps us from making a strong conclusion about the effect of E-Plans on the behaviour of emotional agents.

The remainder of this chapter is organized as follows. We first describe the abstract problem scenarios and the metrics we developed for them. We then present the results and discuss our main observations.

6.2 Experimental Design

6.2.1 Plan Libraries

We defined problem solving scenarios that consisted of three top-level goals, each of which decomposed to one level of sub-goals. These three top-level goals were presented to the agent as three independent goals that could be worked on concurrently, or as three related goals to be solved sequentially. In the sequential case, the agent must first either succeed or fail at "Goal 1" before trying "Goal 2," then succeed or fail at "Goal 2" before trying "Goal 3." For the concurrent case, the agent may work on the goals in any order and may switch between them at any time. Sequential problem solving tells us something about how prior experience impacts how the next (identical) goal is tackled. Concurrent problem solving can reveal how much an agent sticks with a given goal until it succeeds or fails, or whether the agent jumps around from goal to goal (as a function of its emotional state).

Each top level goal had three possible plans. These three plans decomposed into a further level of sub-goals and plans in one of two ways. In the *symmetric* library, each plan had exactly 2 sub-goals of medium value, which in turn had exactly 2 associated plans. Figure 6.1 shows the tree structure of the goals and plans for the symmetric library. Each goal and sub-goal in the figure, represented by ellipses, includes the goal name and goal value G (low, medium, or high). Each plan, represented by boxes, includes the plan name, cost C, and probability of success P (low, medium, or high).

For the *asymmetric* library, a top level goal also had three plans, but these three plans decomposed in different ways as further sub-goals and plans. Figure 6.2 shows the tree structure for the asymmetric library. The library is represented in the same format as in Figure 6.1.

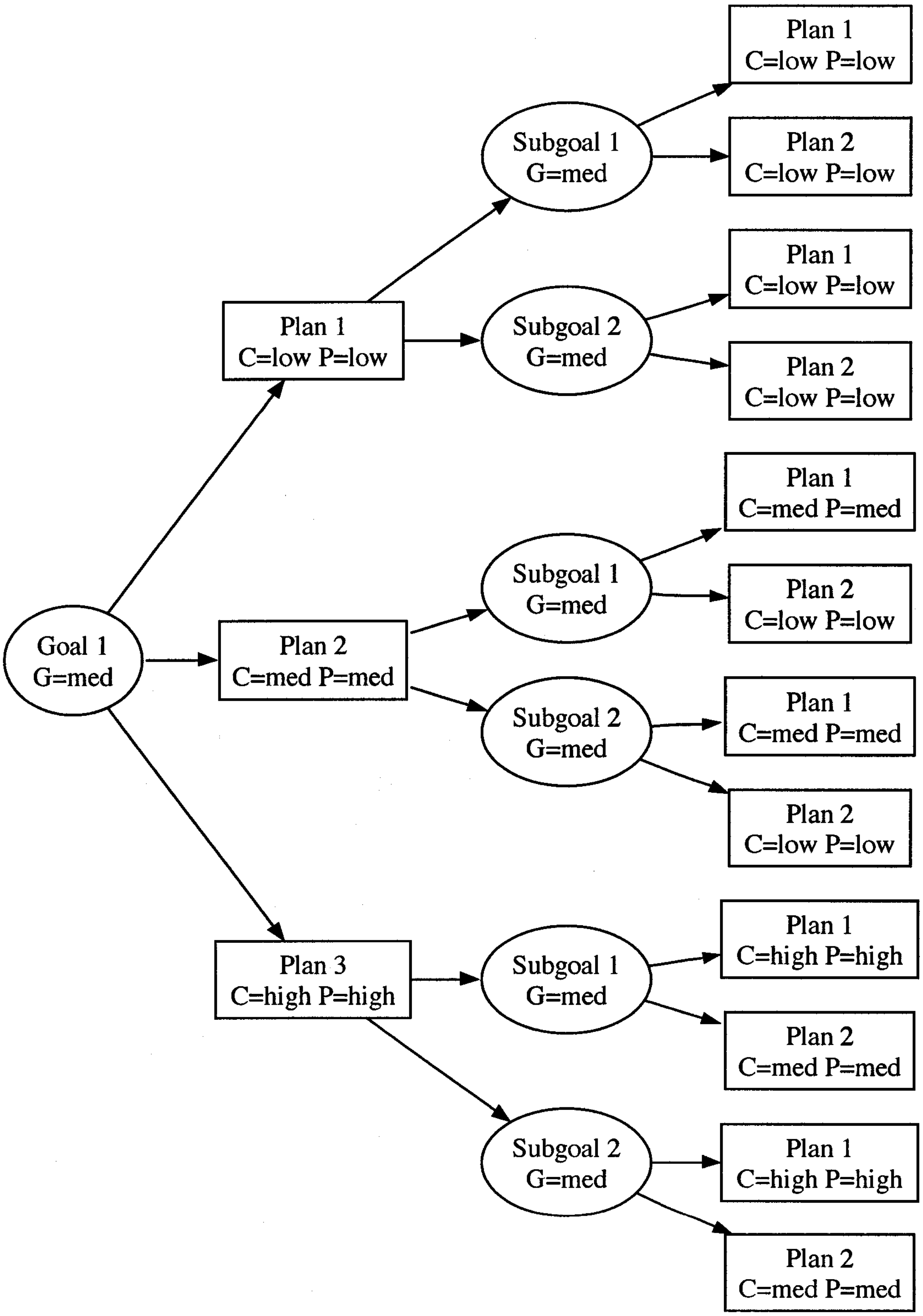

Figure 6.1: Graphical representation of "Goal 1" in the symmetric scenario. "Goal 2" and "Goal 3" have identical structures.

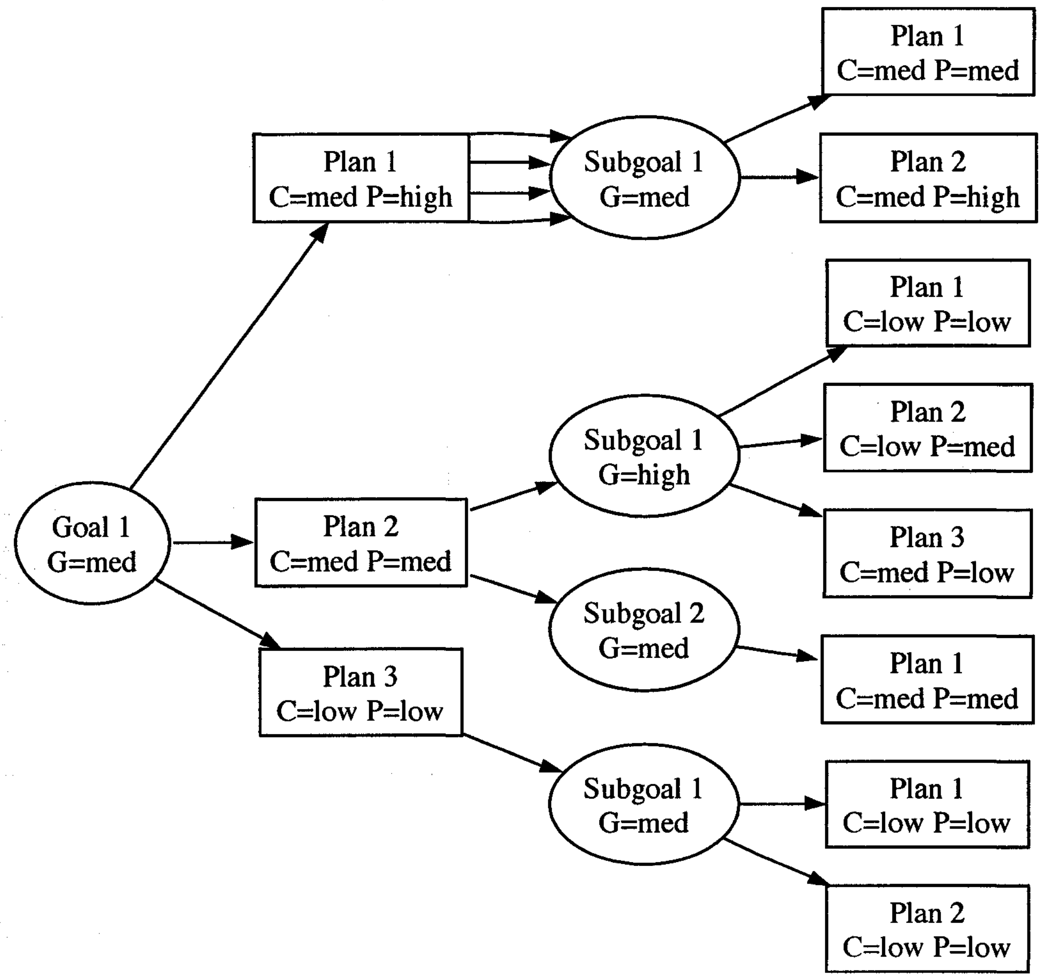

Figure 6.2: Graphical representation of "Goal 1" in the asymmetric scenario. "Goal 2" and "Goal 3" have identical structures.

The rationale behind these two different plan libraries was the following. The symmetric plan library was particularly important for examining how changing emotional state over the course of a problem solving episode changed an agent's problem solving behaviour on the very same goal-plan substructure. This was particularly useful when the three top-level goals were concurrent. The asymmetric plan library, by offering more variability in how plans decomposed, allowed us to define sub-goals that needed to be achieved repeatedly. As can be seen in Figure 6.2, "Plan 1" sub-goals to the same "Subgoal 1" four times. With repeated sub-goals we could test how the agent's plan selection changes when presented with the actual same plans (not just a similar structure) with a different emotional state.

Another variation we considered on the plan libraries is the inclusion of simple homeostasis E-Plans. Plan libraries with E-Plans had one E-Plan for each emotion type, and each E-Plan reduces the value of the associated emotion instance by 0.2. The specified probability of success of each E-Plan is 0.5, though the E-Plans in these experiments actually always succeed.

With the two plan libraries and two possible variation choices given, there are eight "scenarios" in total: asymmetric or symmetric, sequential or concurrent, and with or without E-Plans.

Each scenario is designed so that the failure or success of each plan can be precisely controlled. We chose four different configurations, or problem solving *experiences*, for these experiments. For each experience, we set all plans (and sub-goal plans) of a particular goal to always fail or always succeed. A "success goal" is a top level goal that will always succeed in a particular experience, while a "failure goal" is a top level goal that will always fail. Each scenario has three top level goals. The experiences we chose for these experiments are Failure-Failure-Failure, Success-Success-Success, Failure-Success-Failure, and Success-Failure-Success. Thus, for the Success-Failure-Success experience, "Goal 1" is set to succeed (by the above definition), "Goal 2" fails, and "Goal 3" succeeds. This is true regardless of whether the goals are to be achieved concurrently or sequentially.

6.2.2 Agent Definitions

Four distinct agents are used in the abstract problem experiments. Agents are differentiated by their personality specification. Two of the agents are designed to behave in a "rational" manner, while the other two agents are given an "emotional" personality. We examine each type of agent with noisy and non-noisy variants.

The rational agents have all emotion activation thresholds set to 1.0, so that their emotional state has no impact on their decision-making. One rational agent has its noise variance set to 0, while the other has noise variance 0.05. The purpose of the rational agents is to provide a baseline control for behaviour. The noise variance value was chosen by trial and error so that it has some influence on behaviour but does not overwhelm the decision-making process.

The emotional agents used in these experiments are designed to represent an "average" personality (by the scales we implemented). Activation thresholds are set to the same level (0.3) for all emotion dimensions. This threshold level is set so that emotional effects will be seen after a small number of events without causing the agents to be exceedingly sensitive. As with the rational agents, one emotional agent has zero noise variance, while the other has noise variance 0.05.

6.2.3 Method

The basic method used for these experiments involved running the system with every test combination according to the possible variations of plan libraries, agents, and problem solving experiences. Crossing eight variations of plan libraries with four different agents and four different experiences defines 128 different test combinations. To account for random effects, each test combination was run 100 times. We call each run an *episode*, the result of which is a trace generated by the system that shows the agent's behaviour and emotional state (if applicable). The trace output is a streamlined version of that shown in the example trace in Section 5.9.

The episode traces for each test combination were parsed to retrieve the data we are interested in. The episode data was collected over the 100 episodes for each test combination. We developed three analysis tools that used the data in different ways. Emotion charts use emotional state data averaged over the 100 episodes in each combination for use in investigating the emotional state of an "average" agent. Plan selection counts are a metric for analysing the plan selection behaviour of agents, and can be used to determine how an agent is likely to behave in a given test combination. Goal persistence metrics were computed for concurrent problem solving scenarios, and measure how much agents jump from goal to goal during problem solving. Each of the analysis tools are described in the subsequent sections.

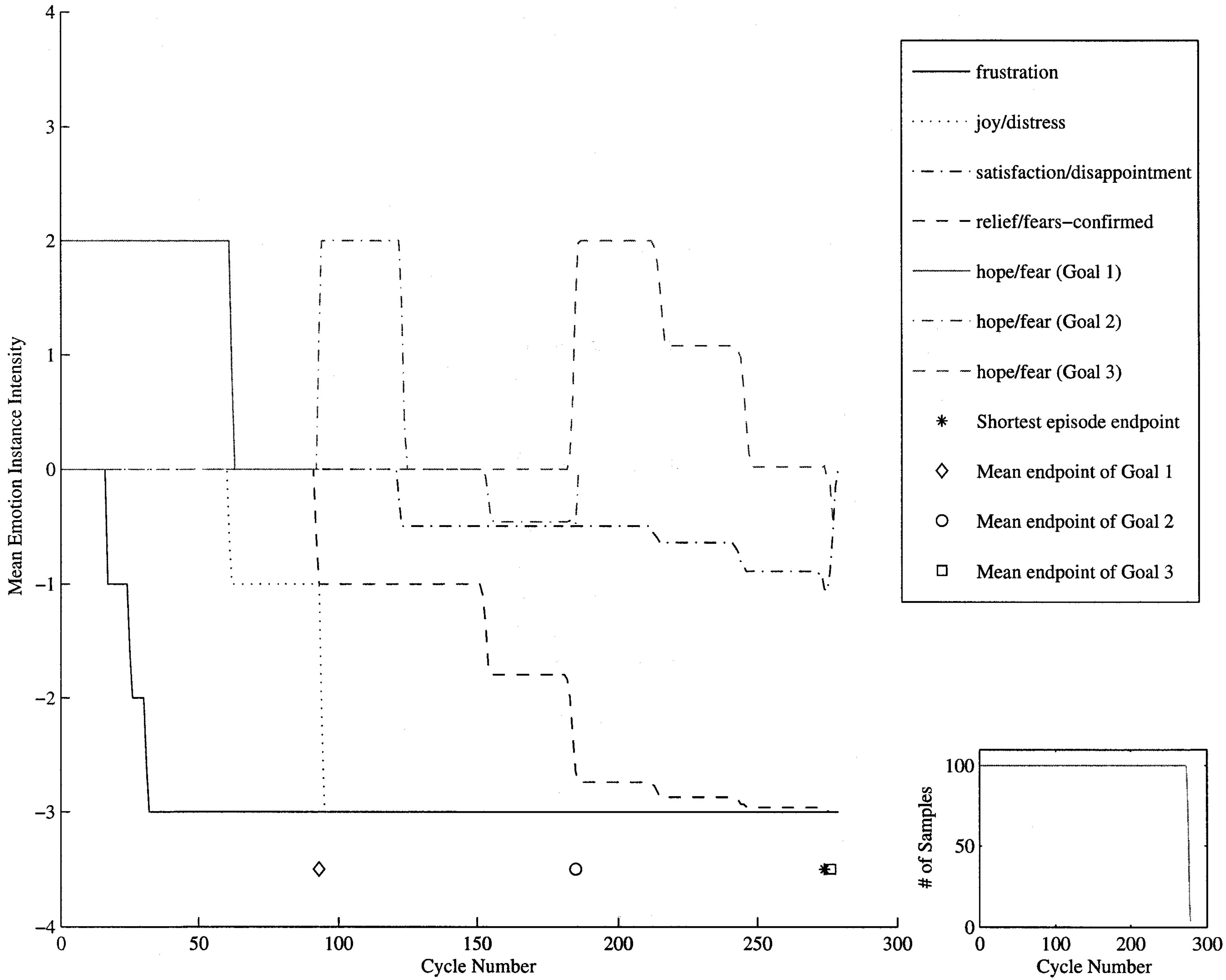

Figure 6.3: Emotion chart for the non-noisy emotional agent. The test combination shown is the symmetric sequential scenario with no E-Plans, for a Failure-Failure-Failure problem solving experience.

6.2.4 Emotion Charts

In order to verify the emotional state changes of agents and to shed light on behavioural impacts, we must analyse the emotional state of the agents. We construct an emotion chart from the averaged emotional state data of each test combination. Each chart shows how the emotional state of an "average" agent changes over time during execution of the test combination. Figure 6.3 shows an example of an emotion chart for a test combination with the Failure-Failure-Failure problem solving experience. The x-axis represents the progression of time by E-JAM cycles, and the y-axis represents the mean world model intensity (from -3 to 3) of emotion instances (recall Table 5.3). The three light grey lines on the chart are associated with the hope/fear emotion instances for the three top-level goals. These lines only appear for the periods of time when the top-level goal is still active. In the example, the average hope/fear for "Goal 1" starts out at medium positive intensity (2), drops to zero intensity, then finally disappears when "Goal 1" succeeds or fails around cycle 100.

The four black lines on the chart indicate the global emotion instances of frustration, joy/distress, relief/fears-confirmed, and satisfaction/disappointment. The emotion chart shows how the average intensity of emotion instances changes over time. For example, in Figure 6.3 all the global emotion instances start out at zero intensity. Since all the plans of the agent are failing, the emotion instances tend toward negative intensity as time progresses. Frustration drops quickly to high negative intensity (-3) before cycle 50, followed by joy/distress around cycle 100. This pattern appears because frustration is affected by plan failure, while joy/distress is affected by goal failure. Since plan failures occur much more often than goal failures, the latter trails the former. The intensity of relief/fears-confirmed drops at a slower rate, reaching high negative intensity (-3) by the end of the test combination. Satisfaction/disappointment also eventually goes to negative intensity, but with a lower magnitude. These emotion instances drop more slowly because they depend on hope/fear being active in certain directions; in this example, there is little hope for the top level goals by the time they fail, so the agent is not often disappointed at that time.

The other features of emotion charts are intended to provide contextual information about the test combination data. The shortest episode endpoint $*$ is indicated because not all episodes for a test combination take the same number of cycles to complete. Any data points after the shortest episode endpoint indicator are aggregated over less than 100 episodes. The small chart in the bottom right hand corner shows in more detail the number of episodes (samples) used to compute each point in the emotion chart. In Figure 6.3 we can see that there is a sharp drop-off of the number of samples, indicating that most of the episodes in this test combination complete in a similar number of cycles. The effect of fewer samples is seen in the erratic jump of the intensity of satisfaction/disappointment at the very end of the test combination.

The mean completion points of each top-level goal are shown by the indicators ◇ (Goal 1), ○ (Goal 2), and □ (Goal 3). In each episode, the completion point for a goal is the cycle number where the goal succeeds or fails. The mean completion point for the test combination is simply the average completion point over all 100 episodes. For example, in Figure 6.3 the mean completion point for "Goal 1" is around cycle 90.

An emotion chart shows what is happening to the emotional state of an average agent in the given test combination. The effect and frequency of events that impact emotional state can be discerned at a high level. For example, frustration is affected by plan failure, so it is affected much more often than emotions affected only by goal failure. We see evidence for this effect in Figure 6.3 because frustration reaches a negative intensity much faster than the other emotion instances. We can also examine how the change in emotional state relates to the mean completion endpoints (success or failure) of the top level goals.

6.2.5 Metrics

We define two metrics to analyse the behaviour of the agents in the experiments. Plan selection counts reveal the decisions made by the agents in all scenarios, and goal persistence statistics indicate how much an agent sticks with a given goal, or jumps to working on a different goal, in the concurrent goal scenarios. For a given scenario, the metrics together along with the emotion charts should tell a consistent story and provide a record and explanation of agent behaviour.

Plan Selection Counts

An emotional agent will compute different plan utilities than a rational agent because of the influence of its emotional state on utility variables. We can observe the impact on decision-making by examining how agents select plans to achieve their goals. A rational agent will always choose the same plans every time, no matter what has happened in the past. An emotional agent, on the other hand, has an emotional state that has been changed by prior events in the world. Its emotional state affects its plan selection behaviour by influencing the utility computation. We measure plan selection behaviour by recording the choices made by an agent over the 100 episodes of a given test combination.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-2-3 (100)	1-2-3 (100)	1-2-3 (100)
Rational (noisy)	1-2-3 (98)	1-2-3 (98)	1-2-3 (98)
Emotional (non-noisy)	1-3-2 (100)	3-1-2 (54)	2-3-1 (23)
		3-2-1 (46)	1-2-3 (19)
Emotional (noisy)	1-3-2 (95)	3-2-1 (37)	1-3-2 (22)
		3-1-2 (34)	3-2-1 (18)

Table 6.1: Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

Our plan selection analysis looks at the plan choices made for top level goals only, in order to get a high level view of plan selection behaviour. As we can see in Figures 6.2 and 6.1, all top level goals have three plans each. This narrows down the field of possibilities for analysis to the different orderings of "Plan 1," "Plan 2," and "Plan 3." For each goal, we count the number of times (out of 100 episodes) each agent chose a particular ordering of plans for that goal. Table 6.1 shows the top plan selection orderings for all the agents in a particular test combination. For each agent, we present the plan orderings on each top level goal. Each ordering is followed by the number of episodes that the agent selected that ordering. For example, the rational non-noisy agent selects the ordering 1-2-3 every time for each goal. However, the emotional non-noisy agent always selects 1-3-2 for "Goal 1," but selects 3-1-2 54 times and 3-2-1 46 times for "Goal 2." By the time the emotional non-noisy agent gets to "Goal 3" it is selecting mostly randomly between the plan orderings, with the top two orderings being 2-3-1 and 1-2-3.

It is important to recognize that the orderings are a record of what the agent selected, and do not represent the agent "pre-planning" its behaviour. For example, the ordering 1-3-2 indicates that the agent first selected "Plan 1" from all three applicable plans. The plan then failed, leaving the agent to select between "Plan 2" and "Plan 3." The agent selected "Plan 3," which failed, leaving it to finally select "Plan 2." All goals with a problem solving experience of "failure" will always have plan selection orderings with three plans, because the agent fails all plans and tries each one once. All "success" goals, however, have plan selection orderings with just one plan. This is because the first plan the agent selects will succeed, causing the goal to succeed. The remaining plans for the goal become irrelevant and cannot be selected. Hence, the data for the Success-Success-Success problem solving experience reflects the choice of only the first plan selected for each goal.

In the sequential cases, the agent first works on "Goal 1," then "Goal 2," then "Goal 3." We can typically see a progressive change in behaviour with the plan selection counts in sequential cases. In the concurrent cases, all goals are being worked on at the same time. As a result of how E-JAM's intention mechanism works, the first plan selection is made for each of the top level goals in the first three cycles of execution. This effect tends to result in plan selection counts being less informative for concurrent cases. We will see an example of this behaviour in Section 6.3.1. Nevertheless, plan selection counts are very useful for revealing the changing behaviour of emotional agents in sequential scenarios. We will use this metric to explore how the agent approaches the "same" problem over time. Since all the top level goals are equivalent, it is the agent's changing emotional state throughout an episode that causes it to behave differently between problem solving experiences (e.g. Failure-Failure-Failure versus Success-Success-Success).

Goal Persistence

In concurrent scenarios, the agent is trying to achieve all three goals (and associated sub-goals) at the same time. It can only focus on a single plan at a time, however; there is no parallel execution. Agents can switch from one plan to another. Since emotional state influences the utility of plans, it is not unreasonable to think that emotional agents might be more likely than rational agents to jump from goal to goal in concurrent scenarios. The hope/fear emotion instances, which influence only a single goal each, may cause an emotional agent to have a lack of persistence on a goal.

In order to measure the persistence of agents in concurrent scenarios, we define a persistence metric as follows. We denote a *sequence* on a top level goal as a series of the following actions: an agent begins execution of a C-Plan, or a C-Plan sub-goals. The C-Plan must be associated with the top level goal or with one of the sub-goals below it. We can say that each action in the sequence represents the agent continuing to work on the top level goal. The *length* of a sequence is the number of consecutive actions associated with the same goal. For example, suppose an agent begins execution of a plan for "Goal 1," which then sub-goals. The agent next begins execution of a plan for that sub-goal. We count that behaviour as three actions, giving a sequence of length 3 (so far). Once the agent takes an action associated with a different top level goal, the sequence for "Goal 1" is finished. E-Plans are not included in sequences, so they are not considered to interrupt a sequence.

With the definition of a sequence in hand, we can define our persistence metric. For a given test combination, we record all sequences in all 100 episodes of the test combination. The *mean persistence* on a goal g is the average length of all the sequences for goal g in the test combination. With our experimental scenarios, this gives us a mean persistence

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.66	5.00	5.50
Rational (noisy)	1.46	1.28	1.47
Emotional (non-noisy)	4.03	2.27	3.91
Emotional (noisy)	2.17	1.21	2.11

Table 6.2: Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

value for each of the three top level goals per test combination. A larger mean persistence for a goal indicates that the agent is usually working on that goal for a long time without interruption, while a small mean persistence indicates that the agent is often switching to another goal. Table 6.2 shows the mean persistence values for agents in a Failure-Success-Failure problem solving experience. We can see that the non-noisy rational agent has the highest mean persistence on all goals.

In our problem solving experiences, success of a top level goal occurs after the agent tries and succeeds at any one plan for each of the sub-goals. Failure of the top level goal, however, occurs only after the agent tries and fails at *all* possible plans for the first sub-goal in each plan. This difference between success and failure experiences leads to a difference in the maximum possible sequence length for a success goal versus a failure goal. For example, consider the symmetric scenario in Figure 6.1. In the successful problem solving experience, the agent could select and execute a top level plan (action 1), achieve "Subgoal 1" (actions 2 and 3), and achieve "Subgoal 2" (actions 4 and 5). This results in a sequence length of 5, which is the maximum possible sequence length in the failure case. In the failure problem solving experience, the agent must try every top level plan, each of which takes 4 actions to fail, leading to a maximum sequence length of 12. The maximum sequence length of the failure experience is significantly different from that of the success experience. This could make it difficult to compare persistence between agents who have different problem solving experiences. For example, the persistence of an agent having all successes may not be directly comparable with the persistence of one who always fails.

To solve this problem, we define a normalized version of the mean persistence metric. We count the total number of actions taken for each goal g over all 100 episodes in a test combination. Dividing these totals by the number of episodes gives us the average total number of actions for each goal g per episode, α_g. For the symmetric scenario, α_g will always be equal to the maximum possible sequence length for g. However, in the asymmetric scenario, α_g may be different from run to run because the sub-goal structure of the plans are

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	0.47	1.00	0.46
Rational (noisy)	0.12	0.26	0.12
Emotional (non-noisy)	0.34	0.45	0.33
Emotional (noisy)	0.18	0.24	0.18

Table 6.3: Normalized mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

all different from one another. For example (see Figure 6.2), success using "Plan 1" takes 9 actions, but success using "Plan 3" takes just 3 actions. α_g represents the highest possible mean persistence the agent could have had for the goal g in the test combination. For this reason, we focus on the symmetric scenario when investigating persistence.

We define the *normalized mean persistence* on a goal g as the mean persistence on goal g divided by α_g. Normalized mean persistence is always between 0 and 1, inclusive. Table 6.3 shows the normalized mean persistence for the same test combination as Table 6.2. The normalized metric allows us to consider direct comparisons between success and failure goals. For example, the normalized mean persistence of the non-noisy rational agent is 0.47 on "Goal 1" and 1.00 on "Goal 2." This gives a very different picture from the non-normalized mean persistence from Table 6.2, which is 5.66 and 5.00 respectively. From this normalized perspective, the non-noisy rational agent is extremely (perfectly) persistent on "Goal 2" and much less persistent on "Goal 1."

We can use the mean persistence metric to analyse the absolute persistence of agents across similar problem solving experiences. To compare agents across different problem solving experiences, we use the normalized mean persistence metric. Both of these metrics allow us to investigate the effects of emotional state, noise, and E-Plans on the persistence behaviour of agents in our experimental scenarios.

6.3 Observations

The experimental design given above leads to a large amount of test combinations and generated data. Testing eight variations of plan libraries with four different agents and four different success configurations adds up to 128 different test combinations. We consider three different analysis tools (emotion charts, plan selection metrics, and mean persistence metrics) for each test combination, giving us 384 data results to analyse. To present all of this data would be very time consuming and overwhelming. Instead, we consider in depth a few major trends that emerged from the data. The supporting results are available in Appendix A.

6.3.1 Emotional agents behave observably differently than rational agents

We included the rational agents in these experiments as a control or baseline with which to compare the emotional agents. Here we examine the difference in behaviour between the non-noisy rational and emotional agents in scenarios without E-Plans.

At the very least, we would expect the emotional agent to exhibit significantly different behaviour from the rational agent. In every test combination the rational agent will show the exact same behaviour, no matter what the outcome of plans. The emotional agent, however, should show changing behaviour as events impact its emotional state. Indeed, the emotional agent should show different behaviour after a success event than after a failure event, because of the different ways that active emotions influence utility computations.

Sequential Scenarios

First we consider behaviour in the sequential case. To compare behaviour with the rational agent in sequential scenarios, we can use only plan selection charts. Emotion charts can be used to explain the behaviour of the emotional agent, but they are by definition not applicable to the rational agent. In the symmetric scenarios, the rational agent always selects top level plans in the order 1-2-3 for each goal. We can see an example of this behaviour in the plan selection counts in Table 6.1. The rational agent behaves this way because without the influence of emotions, the utility of "Plan 1" is always higher than "Plan 2," whose utility is always higher than "Plan 3."

One major trend we see in the symmetric sequential case is that when the emotional agent has a negative emotional state (after failing "Goal 1"), it tends to prefer "Plan 3" for "Goal 2" which has high cost and high probability of success. One effect of frustration is to reduce perceived cost, so with high frustration the high cost of "Plan 3" is disregarded by the emotional agent. In extreme situations of negative emotion, the agent ends up considering all plans as equally viable. This effect occurs because other negative emotions reduce perceived goal value and probability of success, leading to plan utilities at or near zero. The agent selects randomly between applicable plans, acting in a completely irrational manner. Table 6.1 illustrates this phenomenon in the Failure-Failure-Failure problem solving experience. For "Goal 1" the emotional agent first selects "Plan 1" because it its emotional state is neutral (except for hope/fear) and its utility computation is close to the rational agent's. Once "Plan 1" fails, the agent has some frustration, leading it to immediately diverge from

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-2-3 (100)	1-2-3 (100)	1-2-3 (100)
Rational (noisy)	1-2-3 (96)	1-2-3 (99)	1-2-3 (98)
Emotional (non-noisy)	1-3-2 (73)	1-3-2 (76)	1-3-2 (73)
	1-2-3 (27)	1-2-3 (24)	1-2-3 (27)
Emotional (noisy)	1-3-2 (75)	1-3-2 (62)	1-3-2 (71)
	1-2-3 (20)	1-2-3 (31)	1-2-3 (21)

Table 6.4: Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.31	5.00	5.13
Rational (noisy)	1.25	1.25	1.26
Emotional (non-noisy)	4.10	4.24	4.07
Emotional (noisy)	1.76	1.69	1.72

Table 6.5: Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

the rational agent behaviour by selecting "Plan 3" instead of "Plan 2." After "Plan 3" and "Plan 2" fail, "Goal 1" has failed and the agent moves to "Goal 2." At this point the agent has a significantly negative emotional state. The plan selection counts show that the emotional agent always starts with "Plan 3" for "Goal 2," then selects roughly evenly between "Plan 1" and "Plan 2." By the time the agent is working on "Goal 3" it has a very intense negative emotional state and effectively selects randomly between all the applicable plans. Tables giving the plan selection counts for the other problem solving experiences can be found in Appendix A.

Persistence in Concurrent Scenarios

For the concurrent symmetric scenario, we can consider both plan selection counts and persistence metrics to compare the behaviour of rational and emotional agents. As in the sequential case, rational agents always select plans in the order 1-2-3 (see Table 6.4 for the Failure-Failure-Failure problem solving experience). The persistence metrics gathered for the symmetric scenario show that the non-noisy rational agent usually has a mean persistence of slightly over 5 for failure goals and exactly 5 for success goals (see Tables 6.2, 6.5, and Appendix A).

We see very different behaviour for the emotional agent in the concurrent symmetric scenario compared to the sequential scenario. In every problem solving experience, all three goals show the same plan selection pattern. This effect occurs because E-JAM intends plans for all concurrent goals before any events occur. The initial plan selection behaviour for a goal in an individual episode does not depend on the order of goal execution, but the subsequent behaviour does. We see the same preference (as in the sequential scenarios) for "Plan 3" in situations with negative emotional state, but to a lesser degree (see Table 6.4). In most cases for a goal, the agent will first fail "Plan 1" and then select (with a mild or moderate negative state) "Plan 3." However, sometimes the agent will finish working on the other two goals before returning to a goal. By then, the agent could be in a strongly negative emotional state and select "Plan 2" instead of "Plan 3." The persistence metrics for the symmetric scenario reveal that the non-noisy emotional agent is on average less persistent than the non-noisy rational agent. In only a single problem solving experience (Success-Failure-Success) does the non-noisy emotional agent have a mean persistence greater than 5. For most problem solving experiences and goals, the mean persistence is less than 4. For example, we can see this effect in the Failure-Success-Failure and Failure-Failure-Failure problem solving experiences shown in Tables 6.2 and 6.5, respectively. The persistence metrics for other problem solving experiences are given in Appendix A.

By comparing the plan selection and persistence behaviour of rational and emotional agents, we determined that their behaviour is quite different. These results confirm the operation of the system, including emotion appraisal, storage, and influence on utility computation.

6.3.2 Emotional agent behaviour is distinguishable from noisy rational behaviour

The noise component of an E-JAM agent's utility computation could be viewed as a sort of aggregate irrational effect. The factors of goal value, cost, and probability of success can lead an agent to behave in certain well-defined ways, while noise leads the agent to simply behave more randomly. We ran the experiment with noisy and non-noisy variant agents to determine if noise has a significant impact on behaviour, and if noisy rational behaviour can be distinguished from non-noisy emotional behaviour. In other words, we wish to verify that our emotional modelling produces behaviour that is more refined than behaviour obtained by just adding noise to rational behaviour.

One can observe in Tables 6.1 and 6.4 that the noisy rational agent and the non-noisy rational agent differ very little in plan selection behaviour. In fact, this is the case for all variants of the symmetric scenario because the utilities of the top level plans are quite different from one another. The variations in utility computation caused by the noise factor

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-3-2 (100)	1-3-2 (100)	1-3-2 (100)
Rational (noisy)	3-1-2 (57)	3-1-2 (57)	3-1-2 (53)
	1-3-2 (42)	1-3-2 (39)	1-3-2 (47)
Emotional (non-noisy)	1-2-3 (100)	1-2-3 (50)	3-2-1 (21)
		1-3-2 (50)	2-3-1 (20)
Emotional (noisy)	3-1-2 (50)	1-2-3 (35)	1-3-2 (20)
	1-2-3 (36)	1-3-2 (28)	3-2-1 (18)

Table 6.6: Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

are not great enough to significantly modify behaviour. To investigate the effect of noise we can turn to the asymmetric scenario, whose top level plans are closer in utility. Table 6.6 shows the plan selection counts for an asymmetric sequential scenario. We can see that the non-noisy rational agent always first selects "Plan 1," but the noisy agent selects "Plan 3" slightly more often than "Plan 1." This behaviour appears in each test combination for the asymmetric scenario. From these results we can see that noise has a significant effect on the behaviour of rational agents when plan utilities are close enough together.

We consider the effect of noise on the emotional agent for plan selection behaviour. Emotional agents are affected by noise in largely the same way as rational agents, but with an important difference. Emotional state can have an impact on noise, causing the agent to act more or less randomly. From the plan selection counts in Table 6.6, we can see that in the Failure-Failure-Failure problem solving experience, the non-noisy emotional agent always selects the plan ordering 1-2-3 for "Goal 1", while the noisy emotional agent displays more random behaviour. For "Goal 2", we see increasingly random behaviour for both agents, but the noisy agent is significantly more random. By "Goal 3", both of the agents select plans in a completely random fashion. However, by comparing the behaviour of the non-noisy emotional agent to the behaviour of the noisy rational agent, we can conclude that simply adding noise to a rational agent does not reproduce behaviour similar to the emotional agent, particularly in emotional states of moderate intensity. The emotion charts for this test combination (given in Appendix A) show that the emotional states of the noisy and non-noisy emotional agents progress in a similar way to one another. The noisy agent has a smoother curve, indicating that its emotional state is less consistent between episodes than the emotional state of the non-noisy agent. This effect is much more pronounced in the concurrent scenarios.

Another way to analyse the effect of noise is to consider the persistence metrics for con-

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-2-3 (100)	1-2-3 (100)	1-2-3 (100)
Rational (noisy)	1-2-3 (96)	1-2-3 (97)	1-2-3 (95)
Emotional (non-noisy)	1-2-3 (100)	3-1-2 (100)	1-2-3 (100)
Emotional (noisy)	1-2-3 (50)	1-2-3 (46)	1-2-3 (50)
	1-3-2 (48)	3-1-2 (22)	3-1-2 (19)

Table 6.7: Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

current scenarios. For this analysis we use the symmetric concurrent scenario because it has more consistent maximum sequence lengths than the asymmetric scenario. As mentioned in Section 6.3.1, the non-noisy rational agent has a mean persistence of approximately 5 in most cases. When noise is added, the mean persistence drops dramatically to around 1.25 for most cases, including the test combination in Table 6.5. This indicates that noise has a large effect on the persistence of rational agents. Emotional agents see a similar effect, going from non-noisy mean persistence in the 2.5 to 4 range to noisy mean persistence below 2 in most cases. The interesting result here is that the mean persistence of noisy rational agents and noisy emotional agents are not substantially different in each case. This indicates that the effect of noise overshadows the effect of emotional modelling with respect to the persistence metric.

Our results indicate that while noise has a measurable impact on the behaviour of agents, it does not produce behaviour similar to that caused by emotional state. However, when combined together with emotional modelling, noise can be seen to "wash out" the effect of emotions. This effect is not surprising. The fact that some emotions cause perceived noise to increase, especially with negative emotional states, is likely a contributor to this result.

6.3.3 E-Plans cause emotional agents to behave more rationally

Up to this point we have analysed the abstract experimental results from scenarios that do not involve E-Plans. In this section we examine how the inclusion of E-Plans affects the results. Since E-Plans are intended to bring the agent back to a neutral state (the homeostasis model), we would expect that an emotional agent with E-Plans would behave more like the rational agent than would an emotional agent without E-Plans.

When E-Plans are available, the behaviour of the rational agent remains unchanged. This is because E-Plans only become selectable by the agent when emotions are active, and that can never happen with the rational agent. The experimental results confirm that the rational agent's behaviour does not change.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.53	5.00	5.53
Rational (noisy)	1.53	1.29	1.49
Emotional (non-noisy)	2.94	3.11	3.01
Emotional (noisy)	1.51	1.27	1.53

Table 6.8: Mean persistence for the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

To analyse the effect of E-Plans on the behaviour of the emotional agent, we first turn to the plan selection metrics. Table 6.7 shows the plan selection counts for the same scenario as Table 6.1, but with E-Plans included. We can see that the first choice of the emotional agent, "Plan 1" for "Goal 1", is the same as without E-Plans. However, instead of selecting "Plan 3" second, the agent with E-Plans behaves just like the rational agent, selecting "Plan 2" second. For "Goal 2", we see the agent shows evidence of being influenced by negative emotions, as it always starts with "Plan 3." However, unlike the agent without E-Plans, it does not proceed to select randomly between the two remaining plans, but it always selects "Plan 1" second. This again indicates a "return to rationality" caused by E-Plans. This effect is further confirmed by the plan selection behaviour for "Goal 3." The agent without E-Plans selects randomly between all plans each time, but the agent with E-Plans behaves just like the rational agent. We see similar confirmation of the effect of E-Plans in the other scenarios (see Appendix A).

The emotion charts (given in Appendix A) bear out the effect of E-Plans on the emotional state of the agent. Compared to the emotion chart in Figure 6.3, the agent with E-Plans can never build up a negative emotional state for a long period of time. It does show some negative emotional state corresponding to when it makes decisions contrary to the rational agent's behaviour. At that point, the agent is first selecting a plan for "Goal 2," and we saw that it selects "Plan 3" like the emotional agent without E-Plans. An interesting observation from the E-Plan emotion charts is that they show spikes of emotion intensity corresponding to plan and goal failure events. Frustration is associated with plan failure, so its spikes indicate the times when a plan for a top level goal or sub-goal fails and the agent next selects an E-Plan in response to the frustration. Spikes of the other emotion instances (which are affected by goal success and failure) appear less frequently than spikes of frustration.

In the concurrent symmetric scenario, the results are not so clear. In the Success-Success-Success and Success-Failure-Success problem solving experiences, the mean persistence for each goal is increased by the inclusion of E-Plans. In the Failure-Failure-Failure problem solving experiences, the mean persistence for each goal actually decreases. Appendix A provides the data for these cases. For the Failure-Success-Failure problem solving experience, we can compare Tables 6.2 and 6.8. The mean persistence of the non-noisy emotional agent decreases for the failure goals but increases for the success goal. In most cases we found that the persistence on success goals is increased by E-Plans, but the persistence on failure goals is decreased by E-Plans. The persistence metric is highly dependent on the plan selection mechanism of E-JAM. E-Plans are selected like any other plan, and when they are finished the agent selects another plan, resuming a sequence or starting a new one. It is unclear what effect this has on the mean persistence when E-Plans are involved, so we cannot conclusively say whether E-Plans increase or decrease the mean persistence of emotional agents.

Our results lead to mixed conclusions about the effect of E-Plans on the behaviour of emotional agents. Since E-Plans are intended to model the process of homeostasis by returning the agent to a neutral emotional state, we would expect that emotional agents with E-Plans would exhibit behaviour similar to rational agents. We confirmed that E-Plans reduce the emotional state of the agent and return it to a neutral state. Our analysis found that E-Plans cause the emotional agents to select plans more like the rational agents. However, the persistence of agents in concurrent scenarios is not affected consistently by E-Plans. Whether this inconclusive result is a problem or not remains to be seen.

6.4 Detailed Analysis of Individual Episodes

6.4.1 Rationale and Design

The previous set of experiments examined high level patterns in agent behaviour over a large number of episodes. To examine more subtle phenomena, we consider detailed traces of individual episodes using a wider variety of agents and E-Plans. We focus on different styles of emotional agents with different personalities. The trace output is very similar to that shown in the example trace in Section 5.9.

The plan library used in the detailed analysis experiments is based on the sequential symmetric plan library. The simple E-Plans were replaced with a significantly larger and more complex set of E-Plans. Each E-Plan primarily affects a particular emotion type, but also affects the value of all the other emotions of the same valence. The cost and probability of success of the E-Plans vary and each E-Plan has a chance to succeed or fail corresponding to its specified probability of success.

For the detailed analysis we use only emotional agents without noise. The control, or "neutral" agent is the same as the emotional agent without noise given above. Two other non-noisy emotional agents are examined, a "positive" agent and a "negative" agent. The positive agent has a low threshold for positive emotions and a high threshold for negative emotions and could be described as a good-natured, easy-going character. The negative agent has a low threshold for negative emotions and a high threshold for positive emotions, evoking a mean-spirited or grumpy character.

6.4.2 Emotional State Tendencies

The emotion thresholds specified in an agent personality have a significant effect on the emotional state of the agent during an episode. We found that high thresholds lead to a delay of emotion activation (as would be expected). However, once the emotion is finally activated, it is more likely to be at a high intensity level compared to typical intensities seen with low threshold personalities. For example, we see much more high intensity negative emotions in the Failure-Failure-Failure scenario for the positive agent than the negative or neutral agents. The negative and neutral agents have low negative thresholds that lead to a high frequency of low intensity emotion activation. These emotions are continually removed by E-Plans so they do not have a chance to build up to a high intensity.

6.4.3 E-Plan Selection

With a large selection of E-Plans that affect all emotions of a given valence, we would expect to see a more varied E-Plan selection behaviour compared to the basic E-Plans used in the previous experiments. We find that while the E-Plans focused on a specific emotion tend to be selected for that emotion, at times other E-Plans are selected. For example, a highly frustrated agent would usually select an E-Plan focused on frustration, but might choose a different E-Plan because of the influence of frustration on the utility computation. The different agents show significantly different E-Plan selection behaviour given a particular scenario. For example, in the Failure-Failure-Failure scenario, the negative agent selects E-Plans about twice as often as the neutral agent does, and even more often compared to the positive agent. This result makes sense because the negative agent has active emotional state after virtually any failure event, due to its low thresholds for negative emotions. The other agents, with higher thresholds for negative emotions, have fewer opportunities to select E-Plans because a single failure event may not cause their emotional state to become active. The opposite result is seen in the Success-Success-Success case, where there are no failure events. In the Success-Failure-Success and Failure-Success-Failure scenarios, we still see the negative agent selecting the most E-Plans because frustration is activated very often. Positive emotions are only increased by success events, and there are many more failure events in a Failure goal than there are success events in a Success goal.

6.4.4 Utility Computations

With a detailed analysis we can observe how utility computations are made and see why an agent selects one plan instead of others. We consider the initial values of the utility variables as well as their final values (influenced by active emotions). Certain effects are interesting to note with regard to utility computations. Joy has a positive effect on both goal value and probability of success, causing these factors (which are multiplied together) to have a large influence on plan selection when the agent is joyful. This effect is even larger for E-Plans that dissipate joy, because the initial goal value may be determined by the joy intensity in the first place. This goal value is increased further by the utility variable effects of joy, leading to a very high utility for such E-Plans.

Chapter 7

Narrative Experiments

7.1 Introduction

The abstract problem experiments described in the previous chapter were useful to verify the operation of the E-JAM system and evaluate the effects of emotional modelling in a simple problem solving domain. However, they do nothing to answer the question of believability. We would like to know if E-JAM agents behave in a believable fashion given a personality, a world, and events occurring in the world. To explore this question, we decided to try to determine what elements of E-JAM are important for producing believable behaviour. Of course, part of the problem is defining what believability is. Marsella and Gratch developed an alternative evaluation scheme based on measuring the functionality of their system against standard psychological questionnaires [GM04]. With this scheme they avoided the problems associated with defining believability. Reilly, on the other hand, used several direct and indirect questions relating to strength of character and emotions in an attempt to measure believability in an objective way [Rei96]. We take an approach similar to Reilly's by developing a set of questions that are together intended to produce an indicator of the believability of an agent. By presenting narrative story output from E-JAM to human subjects, we explored the importance of four factors: emotional influence on utility, E-Plans, personality, and initial emotional state.

The basic design of the narrative experiments involved generating a set of narrative story traces, presenting them for human subjects to read, and asking the subjects questions about the traces. We added a basic natural language generation facility to E-JAM to support these experiments. The questions asked of the subjects included questions that try to determine the subjects' affinity toward the agent, as well as the subjects' perception of the agent's personality. The details of the experimental design are discussed in Section 7.2.

The narrative experiments are designed to test two main hypotheses about E-JAM agents. While our experimental questions are intended to provide an indicator of believability, we are not asserting that they are proxies for believability. Instead, we are primarily looking for evidence that supports the hypotheses that agents modelled with the E-JAM emotional module are perceived differently than those that are not. The first main hypothesis is that both homeostasis E-Plans and emotional influence on utility affect the responses of subjects to the narrative traces. Furthermore, we hypothesize that E-Plans and emotional influence together have a stronger effect than either do alone. A related minor hypothesis is that using "emotion verbs" (e.g. "shuffles" or "skips" instead of "walks") in narrative traces has an effect on the responses of subjects. The second main hypothesis is that agent personality and initial emotional state both affect the responses of subjects. Specifically, we suggest that an agent with a "negative" (e.g. bad-tempered) personality should be perceived as negative, and that an agent who starts out with a negative emotional state should also be perceived as negative. An agent who has both negative personality and negative emotional state should be perceived as even more negative.

The results of the experiments gave a confirmation of the first main hypothesis. We found that emotional influence on utility and the use of E-Plans both impact the responses of subjects and that their interaction also makes an impact. However, we did not find strong evidence to either confirm or deny the related minor hypothesis concerning emotion verbs. Furthermore, the second main hypothesis was not verified by our results. We found some evidence that supports it, but we also found other evidence that undermines it.

The remainder of this chapter is organized as follows. First, we describe the experimental design in detail. We then present the results of the experiments, followed by a discussion of how the results relate to our hypotheses.

7.2 Experimental Design and Procedure

We developed two experiments to explore the two different main hypotheses. For each experiment we vary two factors, giving a 2-way design. Experiment 1 investigates the effects of the E-JAM emotional module (emotion-model agents vs. rational agents) and of the inclusion of E-Plans (inclusion of E-Plans in the plan library vs. no E-Plans). Recall that the E-JAM emotion module impacts the computation of plan and goal utilities by biasing goal value, probability of success, and cost estimates. The E-JAM emotional module biases these utility components as a function of the emotional state it computes, which in turn is determined by plan and goal success or failure. E-Plans are non-goal related behaviours, with their own value, success probability, and perceived cost, that are effectively triggered by an internal state (i.e. when a particular emotion surpasses its activation threshold). They compete with plans and behaviours dedicated to the achievement of externally-assigned goals.

The purpose of this experiment is to determine if explicit emotional expression (E-Plans) and implicit emotional influence on decision-making are both influential on the responses of subjects (and perhaps on believability). In this experiment, each agent starts with negative emotional state (all emotion instances at -1.0 value). Emotion verbs, which are used in narrative generation to indirectly indicate the agent's emotional state, are not used in Experiment 1.

Experiment 2 explores the effects of personality (neutral vs. negative) and the effects of initial emotional state (neutral vs. negative). This experiment is used to determine whether our modelling of personality and initial state are both important for giving the impression of a negative agent. Effectively, Experiment 2 is geared at a further test of our approach to synthetic agents, since personality settings and initial state effectively determine how soon the emotional module starts biasing utility metrics. In this experiment, emotion verbs and E-Plans are used for all agents.

The materials used for these experiments were narrative traces of agent behaviour in a simple problem solving scenario. The scenario and narrative generation for each experiment was designed so that each narrative trace would take approximately 10 to 15 minutes to read. Subjects were University of Alberta undergraduates who received credit towards a Department of Psychology course requirement for experimental credit. Forty subjects participated in each experiment, with equal numbers of females and males in each experiment. Subjects were run in groups of 5-10 and were native speakers of English.

Subjects were randomly assigned to a particular experimental condition (e.g., emotional agent with no E-plans), which determined the nature of the narrative trace that was produced. There were 10 subjects assigned to each of the four conditions within each experiment.

Each subject received an instruction sheet, followed by a single narrative trace, and a set of questions. The instructions did not indicate that the trace to follow was computer generated, but instead referenced a certain kind of narrative style (see Section 7.2.1 and Appendix B). Subjects worked at their own pace and received debriefing information at the end.

The details of the scenario, agent personalities, narrative generation, and questions are described in the subsequent sections. The full set of experimental materials can be found in Appendix B.

7.2.1 The problem-solving scenario

The scenario requirements for the narrative experiments are very different from the requirements for the abstract problem experiments. The situation and actions of the agent should be interesting and engaging to the reader. These qualities are not provided by either of the scenarios in the abstract problem experiments. Other qualities are important in scenarios that will be used to generate readable narratives. In order to demonstrate how emotional state might evolve from goal and plan outcomes, and in turn start affecting subsequent plan selection decisions, agents need to have multiple ways to accomplish their goals. The scenario also needs to be long enough to demonstrate these behavioural changes, but not too long for a person to read the narrative output in a few minutes. Finally, the scenario needs to be workable and interesting with a single agent.

We developed a narrative problem-solving scenario based on the play "My Foot My Tutor" by Peter Handke [Han77][1]. The play has two characters, the ward and the warden, and involves their wordless interaction throughout a day on a farm. The script is very descriptive and written from the perspective of an observer "noticing" the situation and the actions taking place. There is no dialogue–the audience interprets the characters' state strictly from their behaviour with each other and with objects on the stage. We decided to base our scenario on "My Foot My Tutor" because this very style made it particularly well suited for automatically-generated behaviour traces based on plan outcomes in a simple world. In our scenario, the ward is on the farm by himself, and has a main task to accomplish that has been set by the warden (who does not appear in our narrative traces). The task and the plans to achieve it are inspired by events in the play, as are the E-Plans available to the ward. The following is an excerpt from the opening part of the play as written by Handke.

> The figure on the state is young – some recognize that this figure probably represents the ward.
> The ward has his legs stretched out in front of him.
> We see that he is wearing hobnailed boots.
> The ward is holding the underside of his right knee with his left hand; the right

[1] We gratefully acknowledge many useful discussions with Professor Piet Defraeye, Department of Drama, about the challenges of conveying internal state through simple actions and choices on stage. The choice of this play as inspiration, as well as many features of the experimental questions, are due to his interest and engagement with this project.

> leg, in contrast to the left, is slightly bent.
> We see that the ward is leaning with his back against the backdrop representing the house wall.
> In his right hand the figure is holding a rather large yellow apple.
> Now that the curtain has opened and is open, the figure brings the apple to his mouth.
> The ward bites into the apple, as if no one were watching. The apple does not crunch especially, as if there were no one listening.
> The picture as a whole exudes something of the quality of what one might call profound peacefulness.
> The ward eats the apple, as if no one were watching. [Han77]

Note that these are effectively directions to the stage director and to the actor; the entire play is written in this style.

Since the narrative scenario is an E-JAM plan library, we can represent it graphically in the same way as the abstract scenarios. Figure 7.1 shows the structure of the narrative scenario. The single top level goal for the ward is to change the light bulb in the farmhouse. He has three plans available to achieve the goal. He can find and use a light bulb grabbing device, find and use a ladder, or climb the furniture in the same room. Each plan has a sub-goal for finding a replacement light bulb, and the grabber and ladder plans also have a sub-goal each for finding the grabber or ladder. The ward can try each plan once, and does not have to find a replacement light bulb more than once if a top level plan fails. For example, the ward could decide to change the light bulb using the ladder, find a light bulb, but fail to find the ladder. If he then decides to try climbing the furniture, the sub-goal to find a light bulb is already achieved and he directly proceeds to the rest of the climbing plan.

The narrative scenario can be defined with problem solving experiences (pre-set determinations of successes and failures for particular goals and plans) similar to the abstract problem scenarios. For the narrative experiments, we used a single problem solving experience which results in the ward only succeeding by using the light bulb grabber plan. The replacement light bulb is in the house, and the grabber can be found in the shed. The other two top level plans are pre-set to always eventually fail for the ward. The problem solving experience can include a number of failures, but it is predefined to have the ward eventually succeed by using the light bulb grabber plan to replace the light bulb. The problem solving experience is illustrated in Figure 7.1 with grey elements indicating goals and plans that

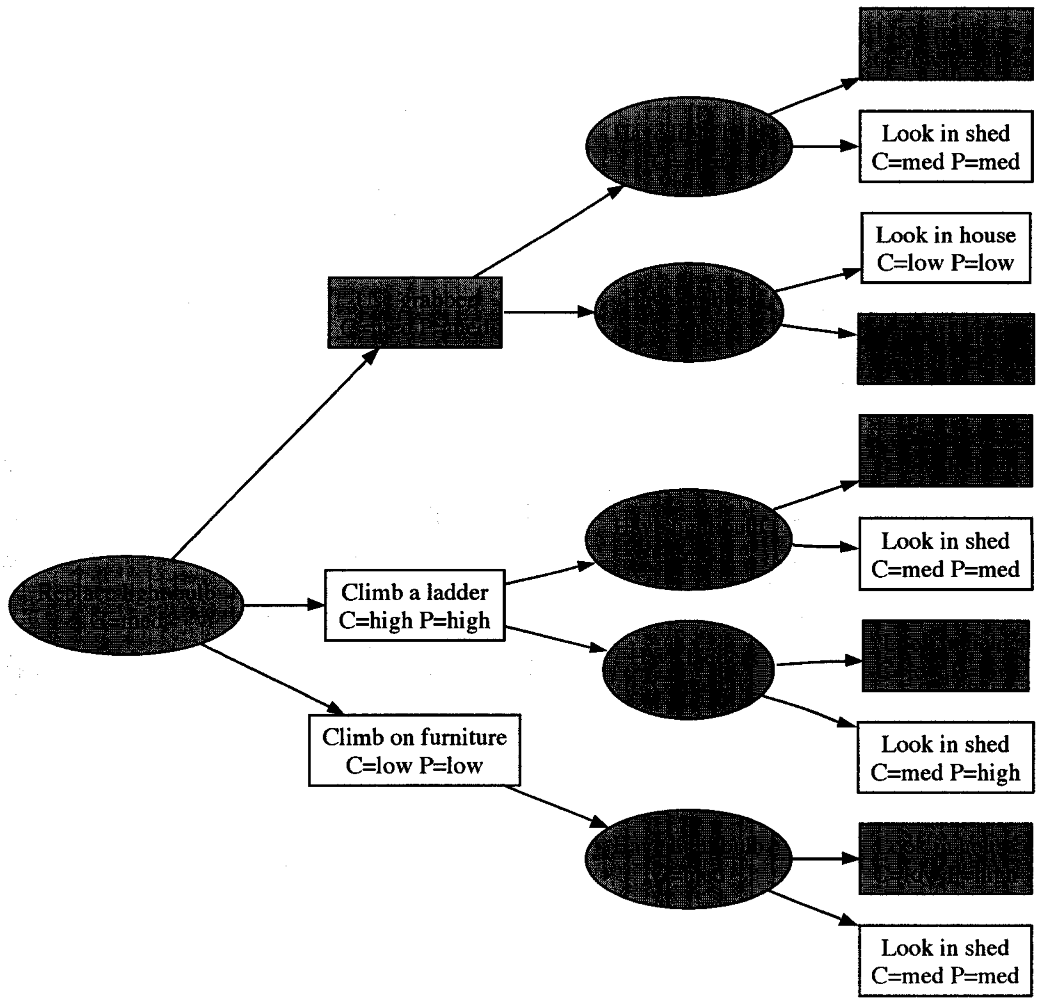

Figure 7.1: Graphical representation of the narrative experimental scenario. Grey ellipses and boxes represent goals and plans that will succeed, and white ellipses and boxes represent goals and plans that will fail.

succeed, and white elements indicating goals and plans that fail.

For experimental conditions testing the influence of E-Plans, the agents plan library contains E-Plans in addition to the cognitive goal achievement plans shown in Figure 7.1. Two E-Plans are triggered by positive emotions and two are triggered by negative emotions. Recall that E-Plans are modelled after the process of homeostasis: they are activated by emotional state, and successful execution of an E-Plan will reduce the level of the emotions that triggered it. All E-Plans compete with C-Plans (cognitive plans for external goal achievement) for the purposes of decision making. The E-Plans are inspired by events in the play, and include drawing in a book (joy-focused), eating an apple (satisfaction-focused), tearing a calendar (distress-focused), and crumpling a newspaper (frustration-focused). Each E-Plan, when successful, reduces the value of the focused emotion by 0.3 and all the other emotions of the same valence by 0.2. If an E-Plan fails, however, it increases the value of frustration by 0.2. For example, the success of the E-Plan for eating an apple reduces satisfaction by 0.3 and reduces joy and relief each by 0.2. E-Plans also have a cost and a probability of succeeding or failing. These values were selected by trial and error to produce traces where E-Plans tended to appear, but did not dominate the narrative. The success or failure of E-Plans are not predetermined like the problem solving experience for C-Plans. The probability of success variable is used to randomly determine whether an E-Plan succeeded or failed.

7.2.2 Agent Personalities

In the narrative experiments we make use of three distinct personalities for the ward agent. The "neutral" agent personality is the same as the non-noisy emotional agent in the abstract problem experiments. It has activation thresholds set to 0.3 for each emotion dimension. In Experiment 1 we explore the effects of emotional influence on utility computations and of E-Plans. In order to cover the case of an agent that has no emotional influence on utility computations but has E-Plans, we need a "rational" agent that can select E-Plans. The rational agent defined in the abstract problem experiments cannot accomplish this, because it never has active emotional state. For the narrative experiments, we define a new rational agent that has emotion activation thresholds set to 0.3 for each emotion dimension, just like the neutral agent. This means that that the emotion module will compute emotional state for the rational agent, as a function of that agents success and failure in the world. We allow this computed emotional state to trigger the possible selection of E-Plans, but do not allow the emotional state to impact the utility computations for any plan or goal that the agent is considering. Effectively, the rational agent here will engage in the same diversity of behaviour (e.g., eating an apple "because" it is joyful) but the state will not cause a biasing of utilities.

The third agent personality used in the narrative experiments is a "negative" personality. The activation thresholds for positive emotions are set to 0.7, while the thresholds for negative emotions are set to 0.1. As a result, the negative agent personality is more reactive to failure situations and less reactive to success situations. The negative agent personality is contrasted with the neutral personality in Experiment 2. The other factor investigated by Experiment 2 is the initial emotional state of the ward agent. Initial emotional state is specified as either neutral or negative. Neutral emotional state has all emotion instance values set to 0.0, while negative emotional state has all emotion instance values set to -1.0 (maximally negative).

7.2.3 Narrative Generation

In order to carry out narrative-based experiments with human subjects, we needed to extend E-JAM to include the ability to generate natural language output. The default output traces, while informative for a researcher, are not conducive to presenting an interesting story to a non-technical user. To support the narrative experiments, we added a simple natural language generation system to E-JAM that includes domain independent and domain specific features. The method of language generation is comparable in some ways to the template system used by Reilly in the Em emotion architecture [Rei96]. Most of the narrative is generated with templates that fill in tags such as *[agent]* with situation or domain specific information (for example, *[agent]* is replaced with "the ward" or "he").

The natural language generation system in E-JAM includes a number of narrative elements that are independent of any specific plan library or scenario. These domain independent elements of narrative generation map neatly to the domain independent aspects of the emotional modelling system itself. Narrative is generated for events (plan failure, goal failure, and success), event and prospect appraisal, sub-goaling, and plan intention and selection.

Event narrative text is different for each event and for C-Plans and E-Plans, and retrieves domain specific information from the plan library as part of the text. For example, the narrative template for C-Plan failure is "*[agent's]* plan for *[plan gerund phrase] [goal infinitive phrase]* has not worked out." In a particular situation, this could result in the narrative output "His plan for looking in the house to get a new light bulb has not worked out."

When an event occurs, the event appraisal process updates the agent's emotional state. For each emotion instance that changes significantly, we output a narrative sentence indicating the change. The sentence includes a reference to the emotion instance using an emotion intensity word. For example, the emotion intensity words for joy/distress (from most distressed to most joyful) are dejected, unhappy, discontented, ambivalent, contented, happy, and ecstatic. The narrative string for the emotion change depends on the nature and direction of the change. For example, if an emotion instance is inactive and becomes active, we simply output "*[agent]* feels *[emotion intensity word]*." However, if the emotion instance was active and its intensity decreased, we would output "*[agent]* now feels only *[emotion intensity word]*." These small differences are important for the flow of the narrative text. Prospect appraisal results in similar narrative generation when hope/fear for a goal changes significantly. The narrative output includes a reference to the goal associated with the hope/fear emotion instance. For example, we would see the narrative output "The ward becomes fearful that he will fail to get a new light bulb" if the agent's fear for that sub-goal increases.

We include domain independent narrative generation for the behavioural choices of the agent. When a plan sub-goals, a narrative template is used to indicate that the agent has to achieve the sub-goal as a step in the plan. The agent's prospect appraisal for the sub-goal is also narrated at this time. We also output narrative text when the agent makes a decision about intending a C-Plan or E-Plan. The intention narration is complex because it includes information about what is being intended as well as the reasons for and against intending it. Furthermore, we include narrative output that mentions both an alternative plan that was not selected and the reason for not selecting it.

The following example shows the three elements discussed above (line numbers and indentation are included for explication and were not in the text seen by subjects).

```
1   The ward plans on looking in the shed
2      in order to get a light bulb grabber
3      because it seems very simple,
4      even though it seems likely to fail.
5
6   He feels paranoid,
7      so he does not care that looking in the shed is actually likely to succeed.
8
9   He had considered looking in the house,
10     but it seems less likely to succeed.
```

The first narrative sentence (lines 1–4) in the example gives the plan the agent has intended and what goal the plan achieves. It may also give a reason (line 3) for intending the plan. The plan was intended because its utility was the highest, so the narrative output gives a reason that the utility is high. Low cost and high probability of success components cause utility to be high, and so we list those utility components as reasons if they have extreme values that improve utility. If they do not, then the reason is omitted entirely. Similarly, a reason against intending the plan (line 4) may also be given if the plan has high cost and/or low probability of success. For example, if there were no reasons for or against intending the plan, the sentence would simply read "The ward plans on looking in the shed in order to get a light bulb grabber." In brief, the reason for plan selection aims to convey elements of the utility computation.

The second sentence (lines 6–7) is generated to explicitly indicate the effects of emotional state on the utility computation biases. In this example, the agent has an active fears-confirmed emotion with high intensity (line 6), causing it to perceive probability of success as lower than it really is (line 7). Finally, the narrative mentions another plan that the agent had considered to achieve the same goal (lines 9–10). If the alternative plan has lower probability of success, lower goal value, or higher cost than the chosen plan, that is given as the reason that the agent did not select the alternative plan. Otherwise, the plans actually have the same utility, and the narrative indicates that the alternative plan was "no better or worse." As in the reasons narrated for plan selection, the reasons for rejecting the alternative plan attempt to reflect information about the utility computations.

In all the natural language generation steps discussed thus far, the only domain specific elements are variables such as plan and goal descriptions drawn from the plan library. In order to tell a more interesting story, we include domain specific narration directly in the plan library that describes the actions taken by the agent. The domain independent narrative can say that the ward decides to look in the house for a light bulb, but it cannot describe the actual act of looking or the reasons for succeeding or failing in the task. For this reason, we include actions directly in each plan that add descriptive text to the narrative. This domain specific narrative text works the same as the domain independent templates to fill in tags such as *[agent]*. For example, when the agent executes the plan to look in the house for a light bulb, the narrative output is "*[agent] [verb=walks]* inside the house and looks around for a light bulb." The *verb* tag is replaced with an emotion verb that is selected according to the highest intensity global emotion instance of the agent. If the agent is disappointed, he shuffles, but if he is joyful, he skips. Emotion verbs are used only in Experiment 2; in Experiment 1, a default verb (e.g. "walks") is always used.

The following is an excerpt from a generated narrative trace. It describes the agent intending a plan to climb the ladder in order to replace the light bulb. The plan sub-goals and the agent decides to look in the shed to find a light bulb. When the plan fails, the agent tries looking in the house, which succeeds.

```
The ward plans on climbing the ladder in order to replace the light bulb because
it seems very simple, even though it seems like a long shot.
The ward feels exasperated, so he does not care that climbing the ladder is
actually difficult.
He feels paranoid, so he does not realize that climbing the ladder is actually
likely to succeed.
He had considered climbing the furniture, but it seems no better or worse.
He goes to work on climbing the ladder to replace the light bulb.
As a step in climbing the ladder to replace the light bulb, he has to get a new
light bulb.
The ward plans on looking in the shed in order to get a new light bulb because it
seems very simple, even though it seems like a long shot.
The ward feels exasperated, so he does not care that looking in the shed is
actually somewhat hard.
He feels paranoid, so he does not realize that looking in the shed is actually
unclear whether it will succeed.
The ward had considered looking in the house, but it seems no better or worse.
The ward walks to the shed and looks inside for a light bulb.
He cannot find a new light bulb in the shed.
He plans on looking in the house in order to get a new light bulb because it seems
 very simple, even though it seems like a long shot.
The ward feels paranoid, so he does not realize that looking in the house is
actually likely to succeed.
He walks inside the house and looks around for a light bulb.
The ward picks up the new light bulb that he finds on the shelf.
```

Due to the basic nature of the narrative generation in E-JAM, and the limited number of experimental traces required, we decided to apply a manual "smoothing" process for each of the traces in the experiments. The smoothing process involves modifying the text in a consistent way to reduce repetitive grammatical structures and improve the variety of word choice. The result of narrative smoothing is a trace that is more readable and engaging. The identical modifications were made to each trace in order to prevent the smoothing from unevenly influencing the experimental results. The following excerpt is the smoothed version of the previous narrative excerpt, which subjects read. The action in the story is the same, but the text has more variety and includes some narrative structures that would be difficult to include in the generated narrative. For example, after the agent fails to find a light bulb in the shed, the smoothed version of the trace reads "*again* we can tell that he thinks this will fail but will be easy." The phrasing of "we can tell" or "we can see" is aimed at assigning an interpretation of the agent's actions to the reader, who would otherwise be viewing an actor on a stage who (presumably) would engage in visual techniques to convey internal state.

```
He plans on climbing the ladder to replace the light bulb.

The ward thinks it will be simple to try.
He is so exasperated that he does not see, as we do, that this is actually going
to be harder than it seems to him.
```

```
He is feeling paranoid too and believes this is very unlikely to work.
We can tell that it probably will work.

He goes to work on his plan.

He first needs to get a new light bulb.

Still exasperated, the ward thinks that looking in the shed will be very easy to
do.
His fearfulness makes it seem to him that finding one there will be a long shot.

He walks to the shed and looks inside for a light bulb.

We see that he does not find one.

The ward now decides on looking in the house.
Again we can tell that he thinks this will fail but will be easy.

Feeling paranoid, he does not see (as we do) that looking there is actually likely
 to work out.

The ward walks inside the house and looks around for a light bulb.

He finds one on the shelf and picks it up.
```

7.2.4 Questions

All subjects answered the same questions that followed whichever trace they read, as per their assignment to a particular experimental condition. The first group of questions includes eight different questions about the ward that are answered by giving a rating on a scale from 1 (not at all) to 7 (very much). We refer to this set of questions as *affinity* questions because they are intended to reveal how the subject feels about the ward and his behaviour. Table 7.1 shows all of the affinity questions. They include simple questions such as "Do you you like the ward?" as well as more philosophical questions such as "Do you think he has a future?"[2] Each question is also accompanied with a prompt for the subject to explain their rating.

1. Do you like the ward?	2. Do you want him to succeed?
3. Do you understand him?	4. Do you identify with him?
5. Do you approve of him?	6. Do his actions make sense to you?
7. Are you on his side?	8. Do you think he has a future?

Table 7.1: The sequence of eight affinity questions used in both of the narrative experiments.

After the affinity questions, each subject was presented with a list of paired adjectives. Each pair contained a positive and negative adjective. The subject was asked to circle the one that he or she thought was the better description of the ward. For example, the subject could circle one of "nice" or "not nice." A total of 13 adjective pairs were presented; the

[2]Again we acknowledge Prof. Defraeye's input on developing these sorts of questions, as alternatives to "Do you regard the agents behaviour as believable?"

nice	not nice
crazy	rational
satisfied	unsatisfied
discontented	happy
satisfied	disappointed
disappointed	relieved
serious	playful
relaxed	frustrated
pessimistic	optimistic
stable	flighty
hardworking	lazy
fearful	hopeful
accepting	intolerant

Table 7.2: The set of thirteen adjective pairs used in both of the narrative experiments.

order of positive and negative pairs was randomly determined, but identical for all subjects. The complete list of adjective pairs is given in Table 7.2.

7.3 Results

The answers to the questions for each experiment were recorded and processed into a data set. The data for the affinity questions are integer values from 1 to 7, inclusive. For the adjective pairs, the data are counts of positive adjectives vs. negative adjectives.

For each experiment, we performed a several 2-way analysis of variance (ANOVA) tests on the data. A separate 2 x 2 ANOVA was done for each separate affinity question, where the data was the value of the subject's scale rating.[3] A separate 2 x 2 ANOVA was done on each adjective pair, where the data was the subject's selection (positive encoded as 1, negative encoded as 2). A 2 x 2 ANOVA was done where the data was the sum of a subject's encoded selections across all adjective pairs (*total adjective selections*). Note that the tests on the adjective pairs are exactly the same in structure as the tests on the affinity questions, except we have data values of 1 or 2 instead of 1 through 7.

7.3.1 Experiment 1

In the first experiment, we consider the factors of emotional influence on behaviour and E-Plans. Each factor has two levels – the emotional modulation of utility factors is present or absent, and the E-Plans as homeostatic responses to emotional state were either available

[3]Ideally, we would also have done a 2 x 2 x 8 ANOVA, with repeated measures on the third factor (question), but for the purposes of this pilot study, we used this simpler analysis.

	Emotional Modulation of Utility			
	Yes		No	
Question	E-Plans	No E-Plans	E-Plans	No E-Plans
1. Like the ward?	3.30	3.40	3.40	3.40
2. Want him to succeed?	4.50	4.40	4.50	5.30
3. Understand him?	3.00	4.10	4.20	3.50
4. Identify with him?	2.10	3.30	4.00	4.10
5. Approve of him?	3.50	3.30	4.00	3.50
6. Actions make sense?	3.60	4.00	4.00	4.60
7. On his side?	3.70	3.65	4.10	3.60
8. Has a future?	3.90	3.60	4.20	4.30

Table 7.3: Mean affinity question rankings for each group in Experiment 1. Each mean is based on 10 subject responses (1 = "not at all," 7 = "very much").

or not for the agent. Table 7.3 shows the average rankings for each question as a function of experimental condition. Recall that on the seven point scale, 1 represents that the subject agrees "not at all" and 7 represents that the subject agrees "very much." We find a significant difference caused by the emotion modulation factor on the rating assigned to Question 4 ("Do you identify with him?") ($F(1,3) = 5.47$, $p < 0.03$). Subjects in the emotion modulation condition gave an average ranking of 2.70, and subjects not in the emotion modulation condition gave a marginal average ranking of 4.05. Subjects identified less with the emotional agent than with the rational agent, which is not necessarily what we expected. We discuss the ramifications of this result and similar results in Section 7.4. We found that for all questions the average ranking for emotional agents is lower than for rational agents (collapsed across the E-Plan factor).

The adjective selection responses are more sensitive than the affinity questions for our manipulations in Experiment 1. Table 7.4 shows the number of positive selections for each adjective pair, as well as the total adjective selections, as a function of experimental condition. Both E-Plans and emotional influence on behaviour have significant effects on the total adjective selections. Additionally, we found six significant effects and two possibly significant effects on individual adjective pairs. We first examine the results for the total adjective selections. Overall, the subjects selected 209 positive adjectives and 310 negative adjectives (40% positive). There were significant main effects for both the emotional modulation factor ($F(1,3) = 8.31$, $p < 0.01$) and the E-Plan factor ($F(1,3) = 4.62$, $p < 0.04$). However, the interaction between the factors is not significant. The agent considered the most positive by subjects is the rational agent with no E-Plans, while the agent considered the most negative is the emotional agent with E-Plans. Recall that the agent was set up to

	Emotional Modulation of Utility			
	Yes		No	
Adjective Pair	E-Plans	No E-Plans	E-Plans	No E-Plans
nice/not nice	5	9	8	8
rational/crazy	5	5	7	7
satisfied/unsatisfied	1	2	3	4
happy/discontented	0	2	2	4
satisfied/disappointed	1	4	3	7
relieved/disappointed	1	7	4	6
playful/serious	0	2	2	3
relaxed/frustrated	1	0	2	8
optimistic/pessimistic	1	2	3	4
stable/flighty	5	3	4	5
hardworking/lazy	5	4	5	4
hopeful/fearful	1	2	5	7
accepting/intolerant	5	7	5*	9
Total adjective selections	31	49	53	76

*This adjective pair had one missing response.

Table 7.4: Positive adjective selection counts for each group in Experiment 1. Each group is based on 10 subject responses, except where indicated.

have multiple failure experiences, and so it is consistent that more subjects select negative adjectives such as frustrated and pessimistic for such agents.

We also see some statistically significant results for certain specific adjective pairs. The most striking result is that for the adjective pair "relaxed/frustrated." We found significant effects for E-Plans ($F(1,3) = 5.49$, $p < 0.03$), emotional influence on utility ($F(1,3) = 17.78$, $p < 0.01$), and the interaction between factors ($F(1,3) = 10.76$, $p < 0.01$). The effect of both E-Plans and emotional influence on utility was to increase the number of selections of "frustrated" instead of "relaxed." The nature of the interaction is that the lack of emotional modulation condition (compared to the emotional modulation condition) increases the number of positive selections more for the agents without E-Plans than for the agents with E-plans. This means that the lack of emotional modulation reinforces the positive effect of the lack of E-Plans. Figure 7.2 shows the effects in a graphical form. For the adjective pair "satisfied/disappointed," similar effects were significant for E-Plans ($F(1,3) = 5.88$, $p < 0.03$) and in the right direction for emotional influence on utility ($F(1,3) = 3.00$, $p < 0.10$). E-Plans have a significant effect ($F(1,3) = 7.38$, $p < 0.02$) for the adjective pair "relieved/disappointed" and an effect in the right direction ($F(1,3) = 3.27$, $p < 0.08$) for the adjective pair "accepting/intolerant." Finally, we found a significant effect for the emotional influence on utility ($F(1,3) = 10.27$, $p < 0.01$) for the adjective pair "hopeful/fearful."

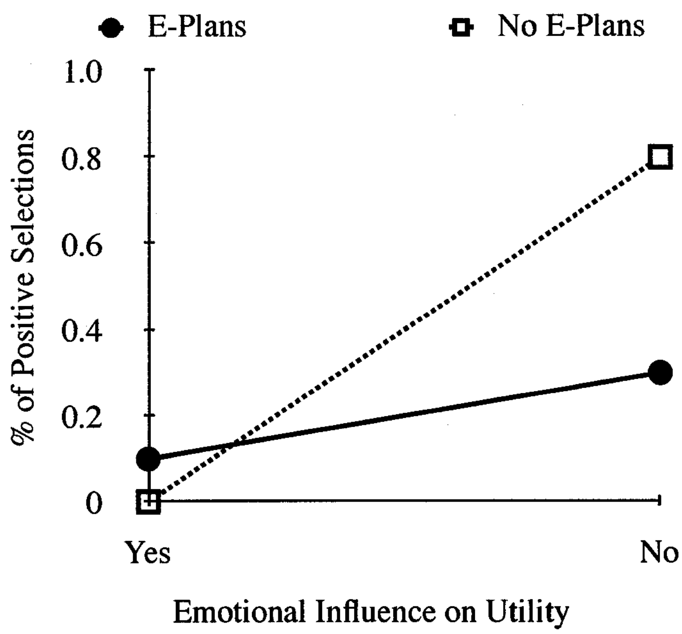

Figure 7.2: Interaction effects in Experiment 1 for the "relaxed/frustrated" adjective selection

In every significant case, agents with E-Plans or emotional influence on utility have a greater number of negative adjective selections. This result was expected because the agents are in a negative situation with negative emotional state and E-Plans and emotional influence on utility cause the agents to "communicate" their negative state to the reader through their actions.

Overall, the affinity questions were not a sensitive measure to our manipulations and this was a disappointment. However, several of the adjective selection questions were sensitive to the conditions of E-Plans and emotional influence on utility. We found evidence that the experimental factors have an effect on the responses of subjects, and that both E-Plans and emotional influence on utility tend to result in a more negative perception of the agents. We discuss these results in more detail in Section 7.4.

7.3.2 Experiment 2

The second experiment explores the effects of initial emotional state versus personality tendencies. As in the first experiment, each factor has two levels. Initial emotional state can be either negative or neutral (zero), and personality can be either negative or neutral. The results are not favourable for the impact of our manipulations, at least with these metrics. Table 7.5 presents the mean ratings for each affinity question as a function of experimental condition. The only notable result is the relative rankings for Question 2 ("Do you want

	Initial Emotional State			
	Neutral		Negative	
Question	Neut Pers.	Neg Pers.	Neut Pers.	Neg Pers.
1. Like the ward?	3.50	3.90	3.40	3.50
2. Want him to succeed?	5.40	4.50	5.00	5.50
3. Understand him?	2.90	3.90	3.30	3.30
4. Identify with him?	3.80	3.40	3.40	3.50
5. Approve of him?	3.70	3.80	3.60	3.90
6. Actions make sense?	3.50	4.50	3.90	4.00
7. On his side?	4.10	3.70	3.90	4.40
8. Has a future?	4.60	4.30	3.90	4.00

Table 7.5: Mean affinity question rankings for each group in Experiment 2. Each group is based on 10 subject responses (1 = "not at all," 7 = "very much").

	Initial Emotional State			
	Neutral		Negative	
Adjective Pair	Neut Pers.	Neg Pers.	Neut Pers.	Neg Pers.
nice/not nice	8*	8	6	10
rational/crazy	7	7	4	7
satisfied/unsatisfied	6	4	0	1
happy/discontented	4	2	0	0
satisfied/disappointed	5*	7	1	3
relieved/disappointed	6	7	5	6
playful/serious	3	0*	0	1
relaxed/frustrated	4	2	2	0
optimistic/pessimistic	5*	5	1	0
stable/flighty	2	7*	1	4
hardworking/lazy	4	7	6	7
hopeful/fearful	6	8	2	1
accepting/intolerant	6	5	4	5
Total adjective selections	66	69	32	45

*This adjective pair had one missing response.

Table 7.6: Positive adjective selection counts for each group in Experiment 2. Each group is based on 10 subject responses, except where indicated.

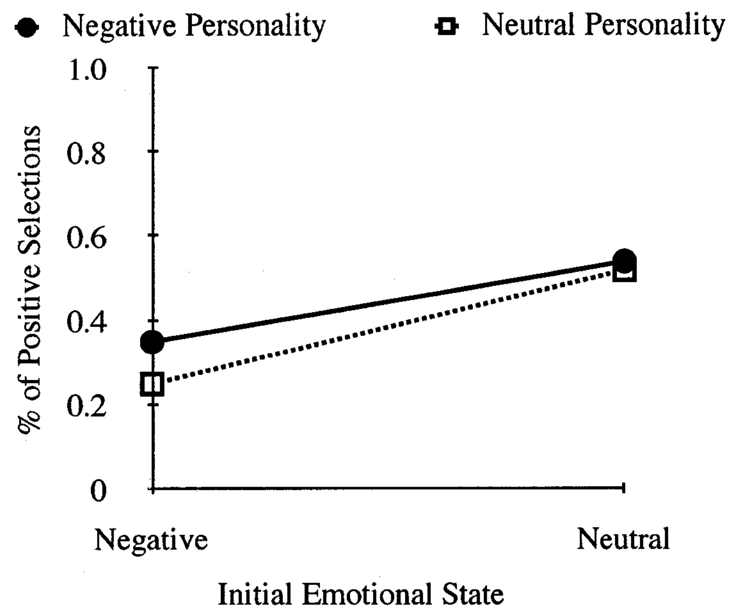

Figure 7.3: Interaction effects in Experiment 2 for the total adjective selections

him to succeed?") (F(1,3) = 2.78, $p < 0.11$). The marginal average ranking on this question for agents with a neutral initial state is 5.0, while the marginal average ranking for agents with a negative initial state is 5.3.

Fortunately, as in the first experiment, the adjective selection results are much more promising. Table 7.6 shows the number of positive selections for each adjective pair, as well as the total adjective selections, as a function of experimental condition. Overall, the subjects selected 212 positive adjectives and 303 negative adjectives (41% positive). The interaction between the personality and initial state factors was significant (F(1,3) = 12.33, $p < 0.01$) for the total adjective selections (see Figure 7.3). The nature of the interaction is that the neutral initial state condition (compared to the negative initial state condition) increases the number of positive selections more for the neutral personality than for the negative personality. This means that neutral initial state reinforces the positive effect of neutral personality. The main effects were not significant.

We also see significant effects for the interaction of both factors on several individual adjective pairs: "satisfied/unsatisfied" (F(1,3) = 12.79, $p < 0.01$), "happy/discontented" (F(1,3) = 8.10, $p < 0.01$), "satisfied/disappointed" (F(1,3) = 8.51, $p < 0.01$), "optimistic/pessimistic" (F(1,3) = 13.83, $p < 0.01$), and "hopeful/fearful" (F(1,3) = 16.75, $p < 0.01$). In two of these adjective pairs we see the same pattern of effect as the total adjective selections, but for the other three pairs we see the opposite effect – the neutral initial state condition decreases the number of positive selections more for the neutral personality than

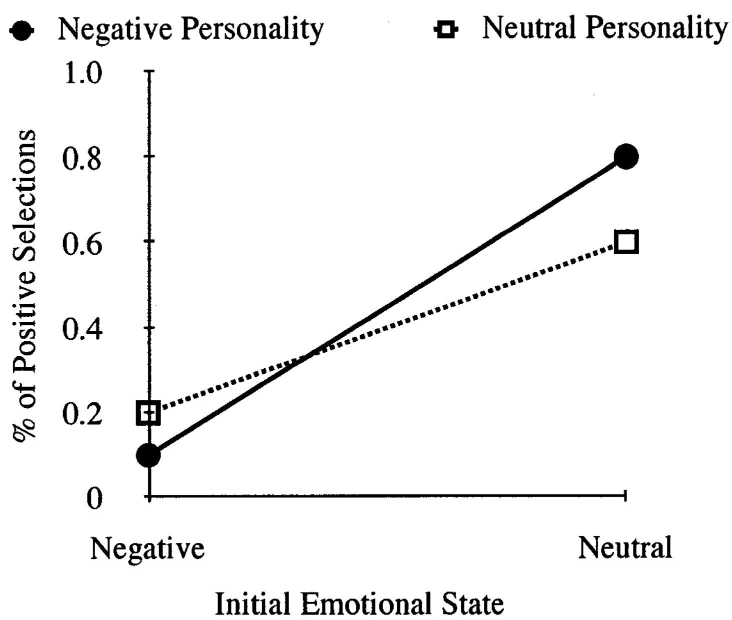

Figure 7.4: Interaction effects in Experiment 2 for the "hopeful/fearful" adjective selection

for the negative personality. This means that neutral initial state works against the positive effect of neutral personality. Figure 7.4 shows the chart for the "hopeful/fearful" adjective selection that reveals the opposite effect (compare with Figure 7.3).

Initial emotional state significantly impacted the "nice/not nice" judgement ($F(1,3) = 4.16$, $p < 0.05$) and the "playful/serious" judgement ($F(1,3) = 4.54$, $p < 0.05$). In both cases, neutral initial state had more positive selections than negative initial state (across personality conditions, see Table 7.6). Note that for "nice/not nice," the sample size of the neutral/neutral group is 9 instead of 10 due to a missing response on one of the questionnaires. The personality factor significantly impacted the "stable/flighty" judgement ($F(1,3) = 10.16$, $p < 0.01$), with neutral personality having fewer positive selections than negative personality (across initial state conditions).

Once again, the affinity questions were not a very sensitive measure. The adjective selection questions revealed significant effects for the interaction of initial emotional state and personality. There is evidence that the experimental factors affect the responses of subjects, but the nature of the effects are inconsistent. We discuss these results in more detail in Section 7.4.

7.3.3 Impact of Emotion Verbs

One plausible concern in any consideration of narrative descriptions of agents is whether language choice accounts for most of the variance in whether an agent is taken to be believable, or whether the reader or observer accords emotional state to the agent. Thus, agent behaviour can be described as He shuffled over to the house or He walked over to the house or He skipped over to the house. Assuming that the emotional state is computed as depressed, neutral, or joyful, respectively, then the choice of a corresponding appropriate verb might influence the perception of the nature of the agent more than any particular problem solving choice per se.

To evaluate the effect of emotion verbs, we compared the judgements of Experiment 1 subjects who judged emotionally modulated narratives with E-Plans with subjects in Experiment 2, who judged neutral personality and negative initial emotional state narratives. Emotion verbs were not used in Experiment 1 and were used in Experiment 2. All subjects in the two groups judged narratives with E-Plans, emotional influence on utility, negative initial emotional state, and neutral personality. The only difference between the two groups is the factor of emotion verbs. The levels of the emotion verbs factor are simply whether or not emotion verbs are used.

We used a one-way ANOVA test to for the cross-experiment analysis, but found very few significant results. The affinity questions revealed no significant effects for the emotion verbs factor. Analysis of the adjective pairs shows effects approaching significance on two pairs, "relieved/disappointed" ($F(1,1) = 4.24$, $p < 0.06$) and "stable/flighty" ($F(1,1) = 4.24$, $p < 0.06$). In the first pair, emotion verbs increase the number of positive selections. However, the opposite effect occurs in the second pair. These results do not give an indication that emotion verbs have a generally discernible effect on the responses of subjects.

7.4 Discussion

The adjective-selection metrics appeared to be more sensitive to our manipulations than the affinity questions. The narrative traces involve an agent experiencing primarily negative emotions and events. The adjective selection questions map particularly well to this situation, because the subject has to make a forced choice between positive or negative adjectives. For example, it is reasonable that an agent who expresses his frustration should have more selections for "frustrated" than an agent who does not express his frustration.

We were disappointed that the affinity questions, borrowed from perspectives on effective acting techniques [Def05], did not prove sensitive to our manipulations. We had hoped that they could serve as an alternative metric to the evaluation of believability. It could be that our manipulations themselves were not strong enough for this metric, or that they were, but the metric and the manipulation are not well suited for each other. It is possible that the number of subjects (10 per cell) was insufficient. The affinity questions were answered with a ranking from 1 to 7, which has no effect of forced choice and may cause a tendency for people to answer around the middle of the range. Increasing the number of observations may address this concern. Finally, we can observe from the written comments from subjects that the interpretation of some questions differed from subject to subject. This could cause two subjects to give potentially wildly different answers to a question for completely different reasons. For example, we consider a selection of responses to the "Do you want him to succeed?" question for the emotional agent with E-Plans in Experiment 1. Some subjects gave a rating of 1 ("not at all") for reasons of story interest, writing that "if he succeeded then there would be no story thus no entertainment," or that "it's more interesting to see a reaction to failure rather than accomplishment." Some other subjects gave a rating of 6 or 7 ("very much") for reasons indicating concern for the agent's physical and mental well-being. One of these subjects would "rather have him just quickly change the light bulb than potentially break his neck due to broken ladders and furniture." Another noted that "it's depressing not to succeed." The written comments to other questions also show that the affinity questions were interpreted in very different ways by different people. This is to be expected to some degree, but the simple and open nature of most of the affinity questions may have led to a wider range of interpretation.

It may seem natural to assume that higher rankings on the affinity questions correspond to increased believability. This may not be the case, considering the negative situation and emotional state of the agents in these experiments. Agents with emotional influence on utility and/or E-Plans may appear to be too negative and overreactive to what happens in the scenario. The affinity questions do not necessarily correspond directly to believability – the closest might be "Do his actions make sense to you?". They may correspond more directly with negative perceptions of the agent. A more positively oriented scenario for the agent could turn the results of the affinity questions around.

We now turn to our earlier hypotheses and consider whether they are validated or invalidated by the experimental results.

7.4.1 E-Plans and emotional influence on utility influence subjects' perception of agents

The first hypothesis is that our model of emotional state, derived from problem solving outcomes and then impacting problem-solving decisions, has an effect on the perception of agents by human subjects. The second hypothesis is that the use of homeostasis E-Plans, behaviours taken in response to internal emotional state, also has a discernible effect on the results. The results of Experiment 1 are mildly supportive of these hypotheses, given our metrics.

The emotional influence factor impacted one affinity question ("Do you want him to succeed?") and two adjective pairing questions ("relaxed/frustrated" and "hopeful/fearful"), as well as the total adjective selections. Each of these questions were characterized by the emotional modulation condition having lower affinity rankings and fewer positive adjective selections. The E-Plan factor impacted three adjective pairing questions ("satisfied/disappointed," "relieved/disappointed," and "relaxed/frustrated"). For each of these questions, agents with E-Plans had fewer positive adjective selections than agents without E-Plans. This was expected because the E-Plans used most often by the agents are for negative emotions, so the E-Plans should give a negative impression of the agents. We also found a significant interaction between the two experimental factors for one adjective pairing ("relaxed/frustrated"). The nature of the interaction was that the lack of emotional influence on utility increased the number of positive selections the most for agents without E-Plans.

The agents in this experiment are in a negative situation (mostly failure) with negative initial emotional state. Furthermore, the negative emotional state of the agents is expressed either by selection and execution of E-Plans, by emotional influence on utility, or both. Therefore, if E-Plans or emotional influence on utility result in agents that are viewed as more negative by subjects, we can also interpret the results as an increase in believability.

Some nuances of the experimental factors should be considered in light of the results. For conditions with emotional modulation, the narrative includes sentences explaining the effect of emotions on the utility computation. For example, consider the sentence "He is so exasperated that he does not see, as we do, that this is actually going to be harder than it seems to him." This sentence is included in order to indicate that there is an emotional influence on the utility computation. The corresponding condition without emotional modulation does not include these explanatory sentences. As a result, the conditions with emotional modulation will have narratives with a larger number of negative adjectives than the conditions without emotional modulation. Additionally, the adjectives "fearful" and "frustrated" appear in the narratives for some conditions. These adjectives are also available for selection in two adjective pair questions.

The fact that both E-Plans and emotional influence on utility impacted a number of questions, all in the same direction, lends support to our hypothesis. The support for their interaction is much smaller, but not insignificant. We conclude that E-Plans and emotional influence on utility do indeed affect the perception of synthetic agents in E-JAM, on some questions. If we consider negative impressions of the agents to indicate some measure of believability, then we can also conclude that E-Plans and emotional influence increase the believability of synthetic agents in E-JAM (again, on some questions).

The related minor hypothesis suggested that emotion verbs also have an effect on the responses of subjects. However, we did not find evidence to confirm or deny this hypothesis. Our results showed that emotion verbs had an impact approaching significance on two adjective pairs. However, emotion verbs increased the number of positive selections on one adjective pair ("relieved/disappointed"), but decreased the number of positive selections on the other ("stable/flighty"). Since the evidence is limited and inconsistent, we cannot make a strong conclusion about the validity of the hypothesis.

7.4.2 Agent personality and initial emotional state influence subjects' perception of agents

The second experiment focused on the hypothesis that negative personality and negative initial emotional state result in negative perceptions of the agent. The consideration of believability in this case is the same as for the previous hypothesis, because the agents are in the exact same situation. If negative personality or negative initial emotional state result in agents that are viewed as more negative by subjects, we could consider the factors effective for believability.

The results for the second experiment are equivocal. The initial emotional state factor impacted two adjective pairing questions ("nice/not nice" and "playful/serious"). Both of the questions showed that neutral initial state results in a higher number of positive selections than negative initial state (the predicted direction). The personality factor impacted a single adjective pairing question ("stable/flighty"), with the effect that neutral personality resulted in a lower number of positive selections than negative personality (the non-predicted direction). We found that personality and initial state did interact to affect judgements on several metrics, including the total adjective selections and five different adjective pairs. Two of the adjective pairs ("satisfied/unsatisfied" and "satisfied/disappointed") and the total adjective selections show that neutral initial state reinforces the positive effect of neutral personality. However, three of the adjective pairs ("optimistic/pessimistic," "hopeful/fearful," and "stable/flighty") show the opposite interaction.

Chapter 8

Summary and Future Directions

We conclude with a discussion of the core elements of this work, its evaluation, and possible future directions. As part of the discussion, we consider how the decisions we made for E-JAM relate to previous research in the field.

8.1 Summary

E-JAM contains several elements that we consider to be core elements of our emotional modelling framework. These four elements are procedural reasoning with C-Plans and E-Plans, the concept of homeostasis applied to emotions, separate utility components with emotional influence on each, and our modified winner-takes-all approach to determining emotional impact on utility. We relate each of these elements to previous research, discuss their strengths and limitations, and consider how well they were tested by our experiments.

8.1.1 Procedural reasoning with C-Plans and E-Plans

Since E-JAM is built upon a procedural reasoning system, we can consider procedural reasoning to be the most central element of all in our approach. An important part of our use of procedural reasoning is that we use the same process to reason about both problem-solving behaviour in the form of C-Plans and emotional behaviour in the form of E-Plans. While procedural reasoning as a specific technique has not been explored heavily in the field of synthetic agents, most emotional models have some notion of goals and plans that concern synthetic agents. The work of Gratch dealt extensively with emotions and planning agents [Gra00], and goals are a central part of the OCC model [OCC88] and the Em architecture [Rei96]. It has been recognized that the concepts of goals and plans are useful for synthetic agents, and our work builds around them as core concepts. By making an investment in using procedural reasoning, we were able to develop a domain-independent emotional modelling architecture on a proven framework.

The limitations of procedural reasoning are carried over into our system, however. Agents are not able to develop new plans on their own, and certain default decision-making aspects of procedural reasoning do not transfer well into synthetic agent behaviour. For instance, as part of the standard procedural reasoning process, agents continually make intentions for goals that do not have assigned plans, even if the agent is not necessarily going to work on the goals right away. This can result in an intention that makes sense at the time, but may not make sense when the agent (in a different emotional state) finally gets around to executing the intended plan. With the flexibility of procedural reasoning architectures, this decision-making process can be modified by designing plans to respond to changes in situation or by using metalevel reasoning to avoid intending plans immediately for each goal.

The domain-independent nature of our architecture was tested by using the same framework for two very different experiments – the abstract problem experiment and the narrative experiment. We developed a completely different plan library and set of agents for each experiment, and ran them on the same framework without problems. The abstract problem experiments also revealed the procedural reasoning decision-making involved with concurrent goals, however. Agents that had several top level goals would intend plans for them all at the beginning of execution, instead of selecting plans at the time they decided to work on a goal. We did not attempt to solve this problem in E-JAM, but a solution was demonstrated by Reilly in the Em architecture. In Em, plan context conditions can be used to cause plans to fail when the agent reaches a certain emotional state. After the plan fails, another plan (with satisfactory emotional preconditions) can be used to achieve the goal. The same concept could be applied to E-JAM plans to reduce the effect of early plan intention.

8.1.2 Homeostasis

Another core element of E-JAM is the use of E-Plans for emotional behaviour. This part of the system is based on the principle of homeostasis, which is the property of a system to regulate itself to maintain a stable state. Whenever an agent's emotional state is no longer in the stable state, it effectively creates a new goal to return to the stable state. E-Plans become available for the agent to execute, and upon success they return the agent's emotional state closer to the stable state. Reilly demonstrated a similar concept in the Em architecture. One method of emotional expression available in Em is the creation of new goals. For instance, the bully character in Reilly's playground scenario will create a new goal to go beat up another character when he gets sufficiently angry at that character [Rei96]. This feature of Em could be used to implement a homeostasis property similar to that of E-JAM. Silverman's work on the effects of stress on synthetic agent includes elements related to homeostasis. An agent under very high levels of stress will only be concerned about its physiological state and will no longer work on its problem solving goals [SJW$^+$02]. This behaviour can be considered similar to homeostasis because the agent focuses only on trying to remove its stress (by running away or cowering). However, it only occurs in extreme cases, does not compete with problem solving behaviour, and does not work with positive emotions.

In E-JAM, we use E-Plans to implement the property of emotional homeostasis as a means of direct emotional expression. One problem with this approach is that E-Plans are not particularly suitable for "small" expressions of emotional state, such as body language or facial expressions. There is a certain amount of overhead involved in intending an E-Plan, selecting it for execution, and executing it. This overhead issue tends to limit E-Plans to more complex modes of expression, such as going for a walk to relieve frustration or throwing a party to express joy. E-Plans can succeed or fail, which is not a notion typically associated with simple emotional expressions like smiling or frowning. Furthermore, since E-Plans compete with C-Plans for the agent's plan selection, and agents cannot execute more than one plan at once, even simple emotional expression may be delayed in an non-believable way. We attempted to mitigate this problem with the introduction of emotion verbs in the second narrative experiment, so that agents could express emotional state through the modulation of the description of simple behaviours (e.g. "shuffling" or "skipping" instead of "walking"). However, we are unaware of other research that directly looks at planning behaviour as a response to emotional state, contributing to a more believable agent. Our notion of homeostasis is that such behaviour itself is likely to impact the level of the emotional state that triggered the behaviour. Of course, that is a separate assumption, and is not necessary for using our framework. One could allow emotions to decay by some other mechanism and also allow include E-Plans that do not impact emotional state, but are conditioned on other factors.

The narrative experiments confirmed that agents using E-Plans were perceived differently from agents not using E-Plans. The agents generally were in a negative emotional state, and in some cases the agents with E-Plans were perceived as more negative than the agents without E-Plans. We also examined the effects of E-Plans in the abstract problem experiments, finding that they cause emotional agents to select plans in a similar way to rational agents. We also confirmed that E-Plans reduce emotional state as designed. However, we did not find that the persistence of agents in working on a given goal was affected consistently by E-Plans in situations where agents could choose between several goals at the same time. It is unclear why E-Plans caused agents to behave in a more rational manner in terms of plan selection, but not in terms of persistence. The persistence metric is highly dependent on the plan selection mechanism of E-JAM, and E-Plans are selected like any other plan. One potential reason for the inconsistency is that when an E-Plan is selected and executed, the agent may still make intentions with active emotional state. Once the E-Plan finishes and its emotional state is reduced, the intentions will have different utility from when they were chosen, possibly resulting in a different selection for execution. However, the exact reason for the inconsistent effect of E-Plans on the persistence metric remains unclear.

8.1.3 Utility computations

A central concept in E-JAM is utility and how it is influenced by emotional state. We use a utility computation drawn from the ACT-R architecture that includes variables for goal value, probability of plan success, plan cost, and noise. We have a strong investment in the idea that different emotions affect the individual variables of the utility computation in different ways. For example, frustration decreases the agent's perception of cost, while hope increases the agent's perception of goal value. The main idea of this concept is that the agent's emotional state has a qualified influence on the decision-making choices made by the agent. Marsella and Gratch integrated emotional modelling with a planning system in their Émile architecture. They focused on the generation of emotional state and the use of coping strategies for emotional expression, but also considered how emotions could affect the nature of problem solving. They suggested that positive emotions could lead to broader problem solving techniques, while negative emotions could cause agents to solve problems in a narrowly focused way [MG01]. Both Belavkin and Silverman, on the other hand, contended that positive emotions are associated with increased motivation and confidence, causing agents to have a narrower focus. Negative emotions would cause agents to select behaviours in a more random fashion [Bel01, SJW^{+}02]. Our own implementation does not constrain the designer to one or the other approaches; by default, however, we tend toward effects similar to those proposed by Belavkin. Reilly also explored how emotional state could influence plan selection by affecting the priority of goals and by requiring emotional preconditions for plans [Rei96]. Effects of this kind are possible in E-JAM, though we did not experiment with using emotional preconditions for C-Plans.

The main challenge to taking our approach to emotional influence on utility is that it involves very subtle effects. Each emotion affects different parts of the utility computation, but the end result is always that some plan is chosen. Given only the choice that was made, an observer may be hard pressed to identify the emotional state of the agent. In a similar vein, the particular nature of the effects were not drawn from a single source. We were unable to find a definitive lexicon that maps emotions to influences on perceptions of cost, value, probability of success, or noise. Some of the effects were drawn from a variety of existing research [Bel01, NITD96, Rei96, SJW+02], but others were based on our own conjecture of a "normal" personality. In sum, the architecture permits an agent designer to decide whether and how different emotional states might impact different elements of utility computations. As a general idea, this could be mapped into more deliberate verbal behaviour. For example, a frustrated agent might say "That way will never work" (perceived probability of success), "That way will take too much effort" (perceived cost), or "That's a stupid thing to want" (goal value). The issue is whether frustration or joy or sadness would be better mapped to one of these elements and not the other, from the perspective either of believability or some kind of consistent character definition.

In the narrative experiments, we set up a test to compare subjects' perception of emotion agents against that of rational agents. Reilly used a similarly structured experiment to test the Em architecture with an emotion-less version of one of his characters [Rei96]. We found that emotional influence on utility makes a difference in subjects' perception of agents. The agents were in a negative situation with negative emotional state, and agents with emotional influence on utility were perceived as more negative in some cases than agents without. In the abstract problem experiments, we found that emotional agents performed observably differently than rational agents on each of our metrics. We also found that agents with emotional influence on behaviour are distinguishable from rational agents with added noise, indicating that the utility effects are more subtle than simple randomness. However, we did find that adding noise to emotional agents tended to wash out the subtleties of the utility computations, making their behaviour similar to that of rational agents (on some personality trait dimensions).

8.1.4 Winner-takes-all approach

Part of the computations for emotional influence on utility involve how active emotions are chosen to influence the utility equation. We used a modified winner-takes-all approach to determine how active emotions are selected to impact on utility. For each component of the utility computation (goal value, cost, probability of success, and noise) we choose the emotion that would have the largest impact on that component. For example, if the agent has a low level of frustration that causes a large impact on cost, frustration will be used to impact cost even if the agent also has a high level of disappointment that causes a smaller impact on cost. Using this method, only one emotion can influence a utility component at a time, but multiple emotions can influence the utility computation overall. Reilly discussed several different methods, including a winner-takes-all-method, for combining the intensities of emotions to result in a single intensity value [Rei96]. The main drawback to the pure winner-takes-all method is that a large number of low intensity emotions will never result in a strong reaction. Our modified method solves this problem by using the emotion with the largest impact on each factor, not simply the emotion with the highest intensity.

One drawback to our modified winner-take-all method for selecting emotional impacts on utility factors is that it exacerbates the difficulty of identifying an agent's emotional state from observations of its behaviour. It is possible, and in fact likely, that more than one emotion will influence a single utility computation. It may not be possible to distinguish between the effects of an emotion that increases goal value and an emotion that increases probability of success, because the two factors are multiplied in the utility equation. Furthermore, the modified winner-take-all method can result in reactions that may be too extreme if a certain mix of emotions result in strong effects on each utility component. For example, if an agent has high satisfaction and high hope, both goal value and probability of success will be increased, while noise is decreased. If we used a winner-takes-all method that selects a single emotion to affect all components, the maximum strength of the impact on utility would be limited by the nature of the impacts of single emotions. For example, relief by itself only increases perceived probability of success; with a standard winner-takes-all method, selecting relief to impact utility would result in all of the other utility components remaining unaffected.

We did not directly test the modified winner-takes-all method in large scale experiments, except as part of the investigations into the effects of emotional influence on utility. However, as part of a detailed analysis of individual episodes in the abstract problem experiments, we observed in detail how utility computations were made by the agents. We verified the modified winner-takes-all method and noted interesting effects, such as the very high utility computations with certain emotional state combinations.

8.2 Evaluation

The results of the abstract problem experiments were generally favourable, except for a few unclear effects related to E-Plans. The experiments confirmed the operation of the framework and the nature of the core elements in an abstract problem solving scenario. However, the narrative user study gave results that were somewhat equivocal. We found that emotional influence on problem solving choices and internally driven plans as expressions of emotional state had an effect on some of the perceptions of subjects, but not as many as we expected. It should be possible to design or redesign narrative experiments in order to convincingly test the core elements of emotional influence on problem solving and homeostasis-driven emotional behaviour. Here we consider two approaches that could improve the evaluation of the E-JAM framework.

Part of the difficulty of designing the narrative experiments was that we needed to generate narrative traces that were long enough to contain enough variation for our tests, but short enough to prevent the reader from becoming distracted or uninterested. For this reason, we decided to have each subject read a single trace of moderate length. Instead of subjects reading a single long narrative (associated with a single experimental condition) that is rated quantitatively on affinity questions and adjectives, we could have subjects read a few short narratives (each associated with a different experimental condition) and rank them qualitatively along some axes. For example, the narratives could be ranked in order of believability, personality traits, and other considerations. This design would likely improve the amount and quality of information we get per subject. The experimental conditions of the proposed design could be similar to those of the original narrative Experiment 1, crossing the factors of emotion modulation and E-Plans. However, a study focusing on the nature of these two elements could use more sophisticated levels than "included" or "not included." For example, we could investigate emotion modulation at half-strength or double strength, E-Plans of varying sensitivity, or E-Plans that do not actually reduce emotional state (perhaps in combination with an ad hoc emotional decay process). By testing a more finely grained spectrum of experimental conditions, we can get a better picture of the nature of the core elements. The main problem with this experimental design is that it makes developing the narrative scenario even more difficult because of the short length requirement for narrative traces.

We gave a considerable amount of thought to including a level of interactivity to our experimental approach. A more interactive experimental design could help to engage subjects in the scenario, giving us higher quality responses at the cost of increased logistical complexity. Including interactivity does not necessarily mean that subjects would interact directly with the synthetic agents. The subject could take the role of director, specifying in broad terms the personality of the agent, then observing how the agent actually behaves. Behind the scenes, we can manipulate the conditions by varying the actual effectiveness of the subject's directions. Since the user is giving directions interactively, the core elements could be modified based on the user's directions. For example, a direction to act angrily could result in lowering of the agent's distress threshold and the inclusion or modification of E-Plans for angry behaviour. The experimental conditions would determine the amount that the threshold would change, for instance. After observing the behaviour of the agent with some specified direction, the subject would answer questions about the agent, or perhaps modify his or her directions and try again to get different results. The data could include what directions the subject gives as the experience goes on. For example, if the subject directed the agent to act happily, and the agent acted in a neutral way in the opinion of the subject (because of the experimental condition), the subject might next direct the agent to act even more happily. This experimental design is very different from the ones we carried out, and so there are several details that would need to be worked out, including improvements in narrative generation and an interactive user interface. However, short episodes of agent behaviour interactively directed by the subject could result in a better evaluation of the core elements of E-JAM.

8.3 Future Directions

We identified several areas of exploration that we did not pursue in E-JAM due to research focus, time, or complexity constraints. Here we discuss five main directions of possible future work, including extension to multiple agents, improvements in personality definition, attention-direction mechanisms, metalevel reasoning capabilities, and graphical 3D interfaces.

8.3.1 Multiple agents

An obvious limitation of E-JAM is that it only supports single agent problem solving situations. This limitation was useful for constraining the complexity of the system and focusing our research by limiting the number of emotion types and personality requirements. However, it also made it difficult for us to develop an engaging narrative scenario for the user study. It is very natural to construct and to read stories about characters interacting with one another as opposed to a single character struggling against him/herself or the environment. Supporting multiple agents opens up a wide area of research, including emotions directed at other agents, modelling the emotional state and personality of other agents, direct interaction with the user as an agent, and so on.

8.3.2 Personality improvements

The personality specification in E-JAM was limited to the definition of activation thresholds, factors for emotional appraisal, and factors for emotional influence on utility. We omitted the specification of standards and preferences, common to many emotion models, in order to focus our research on the core elements discussed above. Including and supporting social standards would be useful for supporting multiple agents, since standards affect how an agent views the actions of itself and other agents in a social context. Agent preferences would allow agents to have more context-sensitive interactions with the environment, including different reactions to events and different evaluations of utility. For example, the utility of a plan to eat an apple could be lower than the utility of a plan to eat an orange if the agent loves oranges, even if the plans are otherwise the same.

8.3.3 Focus of attention

We briefly explored the idea of explicitly modelling a focus-direction mechanism for agents. An emotionally-informed focus of attention model would allow agents to ignore certain stimuli if they are angry, for example. It could also be used to modify the problem solving behaviour of an agent for tasks such as searching. Similar work has been explored by Chopra-Khullar and Badler [CKB99] and by Marsella and Gratch [MG01, MG02] in the context of coping strategies. We decided to focus our research on other areas, but extending E-JAM to support a focus of attention model would be interesting in the future.

8.3.4 Metalevel reasoning

In Chapter 4, we briefly discussed the metalevel reasoning capabilities of the JAM architecture. Metalevel reasoning allows the decision-making process to be defined at the plan level, so that it is possible for reasoning behaviour to compete directly with problem solving or emotional behaviour. For example, a frustrated agent could use different metalevel reasoning plans based on its emotional state, or could even forego any cognitive reasoning at all to focus on a particular task or emotional behaviour. Additionally, making changes to the reasoning process of PRS-style systems is most naturally done with metalevel reasoning, since the system itself does not need to be modified. We investigated the possibility of using metalevel reasoning to implement some of the emotional modelling process in E-JAM, but found that it was far more practical to implement the majority of the emotion module in Java code instead of JAM planning script. We also found that avoiding metalevel reasoning at this time resulted in a more straightforward and understandable system. In the future, it would be worth taking another look at metalevel reasoning to see how it could work with the emotion module to enhance the emotional cognitive reasoning process of E-JAM agents.

8.3.5 Graphical 3D interfaces

We initially explored graphical 3D input and output for E-JAM using the ANIMUS engine developed by Daniel Torres [TB03]. The ANIMUS system includes support for dynamic interpolation between character body positions, giving a potentially interesting avenue for emotional expression. We developed a working interface between our Java agent architecture and the C++/DirectX ANIMUS system that allowed the agent to perceive the 3D world, deliberate and select plans in the agent architecture, and take action in the 3D world. In the end, however, we abandoned this approach in favour of a simpler text-based interface that was easier to develop and evaluate. The potential for future work with graphical integration still exists and could be useful for developing engaging user interactions with agents.

Bibliography

[AB02] J. Allbeck and N. Badler. Toward representing agent behaviors modified by personality and emotion. In *Workshop: Embodied Conversational Agents at AAMAS'02*, Bologna, Italy, 2002. ACM Press.

[And93] J. R. Anderson. *Rules of the Mind*. Lawrence Eribaum Associates, Hillsdale, NJ, 1993.

[Bel01] R. V. Belavkin. The role of emotion in problem solving. In C. Johnson, editor, *Proceedings of the AISB'01 Symposium on Emotion, Cognition and Affective Computing*, pages 49–57, Heslington, York, England, 2001.

[BL80] I. Bartenieff and D. Lewis. *Body Movement: Coping with the environment*. Gordon and Breach, New York, 1980.

[BLR92a] J. Bates, A.B. Loyall, and W.S. Reilly. An architecture for action, emotion, and social behavior. In *Proceedings of the 4th European Workshop on Modeling Autonomous Agents in a Multi-Agent World*, pages 55–68, S. Martino al Cimino, Italy, July 1992.

[BLR92b] J. Bates, A.B. Loyall, and W.S. Reilly. Integrating reactivity, goals, and emotion in a broad agent. In *Proceedings of the 14th Annual Conference of the Cognitive Science Society*, pages 696–701, Bloomington, IN, July 1992.

[CKB99] S. Chopra-Khullar and N. Badler. Where to look? automating attending behaviors of virtual human characters. In *Proceedings of Autonomous Agents '99*, pages 16–23, Seattle, 1999.

[Def05] P. Defraeye. personal communication, 2005.

[Ell92] C. Elliott. *The Affective Reasoner: A process model of emotions in a multi-agent system*. PhD thesis, Northwestern University, 1992.

[GM04] J. Gratch and S. Marsella. Evaluating a general model of emotional appraisal and coping. In *Architectures for Modeling Emotion: Cross-Disciplinary Foundations: Papers from 2004 AAAI Spring Symposium*, Mar 2004.

[Gra00] J. Gratch. Émile: Marshalling passions in training and education. In C. Sierra, M. Gini, and J. S. Rosenschein, editors, *Proceedings of the 4th International Conference on Autonomous Agents*, pages 325–332, Barcelona, Catalonia, Spain, 2000. ACM Press.

[Han77] P. Handke. My foot my tutor. In *Shakespeare the Sadist and other plays*, pages 55–73. Eyre Methuen, London, 1977.

[Hub99] M. J. Huber. JAM: A BDI-theoretic mobile agent architecture. In *Proceedings of the 3rd International Conference on Autonomous Agents*, pages 236–243, Seattle, WA, 1999.

[Hud04] E. Hudlicka. Beyond cognition: Modeling emotion in cognitive architectures. In *Proceedings of the 6th International Conference on Cognitive Modeling*, pages 118–123, 2004.

[JS99] O. P. John and S. Srivastava. The big five trait taxonomy: History, measurement, and theoretical perspectives. In L. A. Pervin and O. P. John, editors, *Handbook of personality: Theory and research*, pages 102–138. Guilford, New York, 2nd edition, 1999.

[MG01] S. Marsella and J. Gratch. Modeling the interplay of emotions and plans in multi-agent simulations. In *Proceedings of the 23rd Annual Conference of the Cognitive Science Society*, pages 622–627, 2001.

[MG02] S. Marsella and J. Gratch. A step towards irrationality: Using emotion to change belief. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems*, pages 334–341, Bologna, Italy, Jul 2002.

[Min74] M. Minsky. A framework for representing knowledge. Technical Report 306, MIT-AI Laboratory, 1974.

[MJL00] S. Marsella, W. L. Johnson, and C. Labore. Interactive pedagogical drama. In C. Sierra, M. Gini, and J. S. Rosenschein, editors, *Proceedings of the 4th International Conference on Autonomous Agents*, pages 301–308, Barcelona, Catalonia, Spain, 2000. ACM Press.

[Mye97] K. L. Myers. User guide for the procedural reasoning system. Technical report, Artificial Intelligence Center, SRI International, Menlo Park, CA, 1997.

[NITD96] T. E. Nygren, A. M. Isen, P. J. Taylor, and J. Dulin. The influence of positive affect on the decision rule in risk situations: focus on outcome (and especially avoidance of loss) rather than probability. *Organizational Behavior and Human Decision Processes*, 66(1):59–72, Apr 1996.

[OCC88] A. Ortony, G. L. Clore, and A. Collins. *The Cognitive Structure of Emotions*. Cambridge, 1988.

[PG96] K. Perlin and A. Goldberg. Improv: A system for scripting interactive actors in virtual worlds. *Computer Graphics*, 30(Annual Conference Series):205–216, 1996.

[PI02] H. Prendinger and M. Ishizuka. Scripting the bodies and minds of life-like characters. In *Proceedings of the 7th Pacific Rim International Conference on Artificial Intelligence (PRICAI-02)*, pages 571–580, 2002.

[Rei96] W. S. Reilly. *Believable Social and Emotional Agents*. PhD thesis, Carnegie Mellon University, 1996.

[RHR97a] D. Rousseau and B. Hayes-Roth. Improvisational synthetic actors with flexible personalities. Technical Report KSL 97–10, Stanford University, 1997.

[RHR97b] Daniel Rousseau and Barbara Hayes-Roth. A social-psychological model for synthetic actors. Technical Report KSL97–07, Stanford University, 1997.

[RN95] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, New Jersey, 1995.

[SJW+02] B. G. Silverman, M. Johns, R. Weaver, K. O'Brien, R. Silverman, and J. Cornwell. Using human behavior models to improve the realism of synthetic agents. *Cognitive Science Quarterly*, pages 273–301, 2002.

[TB03] D. Torres and P. Boulanger. The ANIMUS Project: A framework for the creation of interactive creatures in immersed environments. In *Proceedings of ACM VRST 2003*, pages 91–99, Osaka, Japan, 2003.

[Vel97] J. Velásquez. Modeling emotions and other motivations in synthetic agents. In *Proceedings of AAAI-97*, pages 10–15, 1997.

[Vel98] J. Velásquez. When robots weep: Emotional memories and decision-making. In *Proceedings of AAAI-98*, pages 70–75, 1998.

Appendix A

Abstract Problem Experiment Data

This appendix includes the results from the abstract problem experiments discussed in Chapter 6. Section A.1 presents the results for the symmetric scenario and Section A.2 covers the asymmetric scenario. Each section includes a full set of plan selection counts, emotion charts, and mean persistence data.

A.1 Symmetric Scenario

A.1.1 Plan Selection Counts

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-2-3 (100)	1-2-3 (100)	1-2-3 (100)
Rational (noisy)	1-2-3 (98)	1-2-3 (98)	1-2-3 (98)
	1-3-2 (2)	1-3-2 (1)	2-1-3 (2)
Emotional (non-noisy)	1-3-2 (100)	3-1-2 (54)	2-3-1 (23)
		3-2-1 (46)	1-2-3 (19)
Emotional (noisy)	1-3-2 (95)	3-2-1 (37)	1-3-2 (22)
	1-2-3 (2)	3-1-2 (34)	3-2-1 (18)

Table A.1: Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1 (100)	1 (100)
Rational (noisy)	1 (98)	1 (98)	1 (97)
	2 (2)	2 (2)	2 (3)
Emotional (non-noisy)	1 (100)	1 (100)	1 (100)
Emotional (noisy)	1 (98)	1 (94)	1 (81)
	2 (2)	2 (6)	2 (19)

Table A.2: Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1-2-3 (100)	1 (100)
Rational (noisy)	1 (99)	1-2-3 (95)	1 (99)
	2 (1)	2-1-3 (4)	2 (1)
Emotional (non-noisy)	1 (100)	1-3-2 (100)	3 (100)
Emotional (noisy)	1 (97)	1-3-2 (93)	3 (64)
	2 (3)	2-3-1 (4)	2 (29)

Table A.3: Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-2-3 (100)	1 (100)	1-2-3 (100)
Rational (noisy)	1-2-3 (95)	1 (98)	1-2-3 (97)
	1-3-2 (3)	2 (2)	1-3-2 (3)
Emotional (non-noisy)	1-3-2 (100)	3 (100)	1-2-3 (50)
			1-3-2 (50)
Emotional (noisy)	1-3-2 (95)	3 (64)	1-2-3 (50)
	1-2-3 (5)	2 (23)	1-3-2 (50)

Table A.4: Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-2-3 (100)	1-2-3 (100)	1-2-3 (100)
Rational (noisy)	1-2-3 (96)	1-2-3 (97)	1-2-3 (95)
	1-3-2 (2)	2-1-3 (3)	2-1-3 (4)
Emotional (non-noisy)	1-2-3 (100)	3-1-2 (100)	1-2-3 (100)
Emotional (noisy)	1-2-3 (50)	1-2-3 (46)	1-2-3 (50)
	1-3-2 (48)	3-1-2 (22)	3-1-2 (19)

Table A.5: Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1 (100)	1 (100)
Rational (noisy)	1 (98)	1 (96)	1 (96)
	2 (2)	2 (4)	2 (4)
Emotional (non-noisy)	1 (100)	1 (100)	1 (100)
Emotional (noisy)	1 (95)	1 (84)	1 (83)
	2 (5)	2 (13)	2 (17)

Table A.6: Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1-2-3 (100)	1 (100)
Rational (noisy)	1 (97)	1-2-3 (98)	1 (98)
	2 (3)	1-3-2 (1)	2 (2)
Emotional (non-noisy)	1 (100)	1-2-3 (100)	3 (100)
Emotional (noisy)	1 (100)	1-2-3 (52)	1 (50)
		1-3-2 (35)	3 (33)

Table A.7: Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-2-3 (100)	1 (100)	1-2-3 (100)
Rational (noisy)	1-2-3 (97)	1 (97)	1-2-3 (97)
	2-1-3 (3)	2 (3)	2-1-3 (2)
Emotional (non-noisy)	1-2-3 (100)	3 (100)	1-2-3 (100)
Emotional (noisy)	1-2-3 (56)	1 (56)	1-2-3 (71)
	1-3-2 (40)	3 (34)	1-3-2 (22)

Table A.8: Plan selection counts (out of 100 episodes) for the symmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-2-3 (100)	1-2-3 (100)	1-2-3 (100)
Rational (noisy)	1-2-3 (96)	1-2-3 (99)	1-2-3 (98)
	1-3-2 (4)	2-1-3 (1)	2-1-3 (2)
Emotional (non-noisy)	1-3-2 (73)	1-3-2 (76)	1-3-2 (73)
	1-2-3 (27)	1-2-3 (24)	1-2-3 (27)
Emotional (noisy)	1-3-2 (75)	1-3-2 (62)	1-3-2 (71)
	1-2-3 (20)	1-2-3 (31)	1-2-3 (21)

Table A.9: Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1 (100)	1 (100)
Rational (noisy)	1 (100)	1 (100)	1 (99)
			2 (1)
Emotional (non-noisy)	1 (100)	1 (100)	1 (100)
Emotional (noisy)	1 (91)	1 (96)	1 (93)
	2 (9)	2 (4)	2 (7)

Table A.10: Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1-2-3 (100)	1 (100)
Rational (noisy)	1 (100)	1-2-3 (99)	1 (99)
		2-1-3 (1)	2 (1)
Emotional (non-noisy)	1 (100)	1-3-2 (100)	1 (100)
Emotional (noisy)	1 (92)	1-3-2 (68)	1 (94)
	2 (8)	1-2-3 (27)	2 (6)

Table A.11: Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-2-3 (100)	1 (100)	1-2-3 (100)
Rational (noisy)	1-2-3 (96)	1 (100)	1-2-3 (95)
	1-3-2 (3)		1-3-2 (4)
Emotional (non-noisy)	1-3-2 (83)	1 (100)	1-3-2 (82)
	1-2-3 (17)		1-2-3 (18)
Emotional (noisy)	1-3-2 (76)	1 (95)	1-3-2 (74)
	1-2-3 (19)	2 (4)	1-2-3 (22)

Table A.12: Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-2-3 (100)	1-2-3 (100)	1-2-3 (100)
Rational (noisy)	1-2-3 (97)	1-2-3 (96)	1-2-3 (96)
	1-3-2 (3)	1-3-2 (3)	1-3-2 (2)
Emotional (non-noisy)	1-2-3 (100)	1-2-3 (100)	1-2-3 (100)
Emotional (noisy)	1-2-3 (84)	1-2-3 (82)	1-2-3 (75)
	1-3-2 (9)	1-3-2 (11)	1-3-2 (18)

Table A.13: Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1 (100)	1 (100)
Rational (noisy)	1 (100)	1 (100)	1 (97)
			2 (3)
Emotional (non-noisy)	1 (100)	1 (100)	1 (100)
Emotional (noisy)	1 (94)	1 (89)	1 (94)
	2 (6)	2 (11)	2 (6)

Table A.14: Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1-2-3 (100)	1 (100)
Rational (noisy)	1 (99)	1-2-3 (98)	1 (99)
	2 (1)	1-3-2 (2)	2 (1)
Emotional (non-noisy)	1 (100)	1-2-3 (100)	1 (100)
Emotional (noisy)	1 (92)	1-2-3 (77)	1 (93)
	2 (6)	1-3-2 (16)	2 (6)

Table A.15: Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-2-3 (100)	1 (100)	1-2-3 (100)
Rational (noisy)	1-2-3 (94)	1 (99)	1-2-3 (98)
	1-3-2 (4)	2 (1)	2-1-3 (2)
Emotional (non-noisy)	1-2-3 (100)	1 (100)	1-2-3 (100)
Emotional (noisy)	1-2-3 (78)	1 (92)	1-2-3 (75)
	1-3-2 (18)	2 (8)	1-3-2 (15)

Table A.16: Plan selection counts (out of 100 episodes) for the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

A.1.2 Emotion Charts

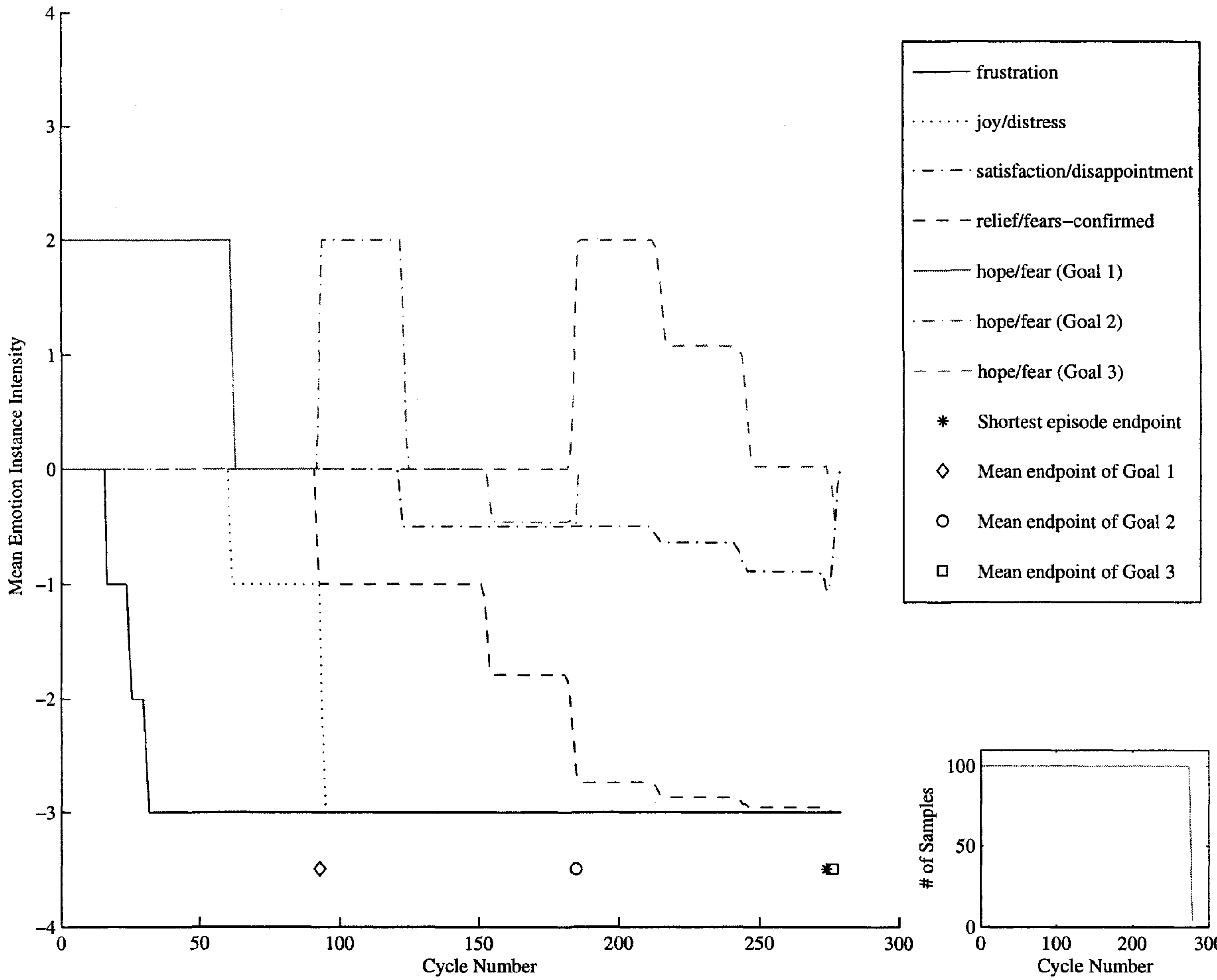

Figure A.1: Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

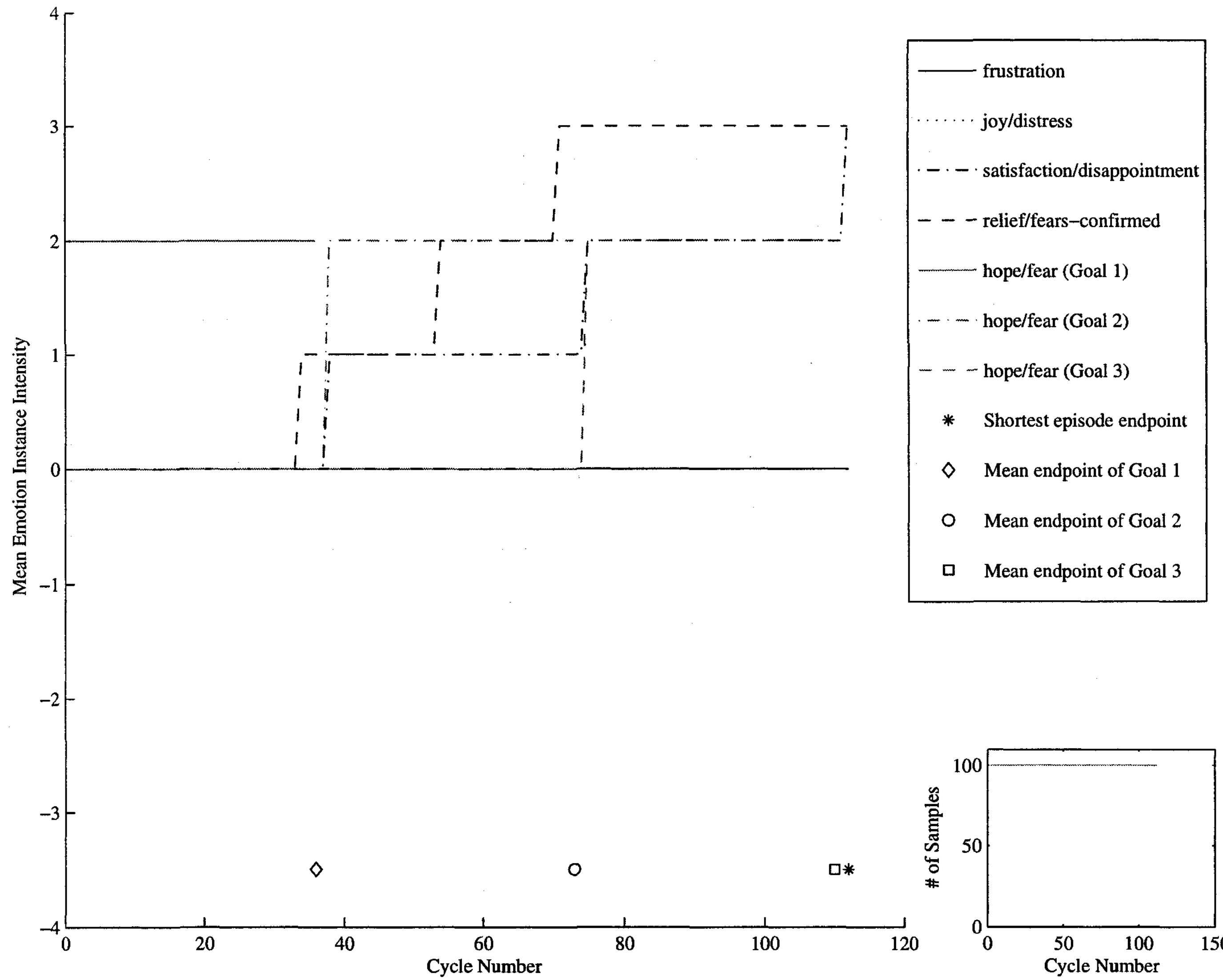

Figure A.2: Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience.

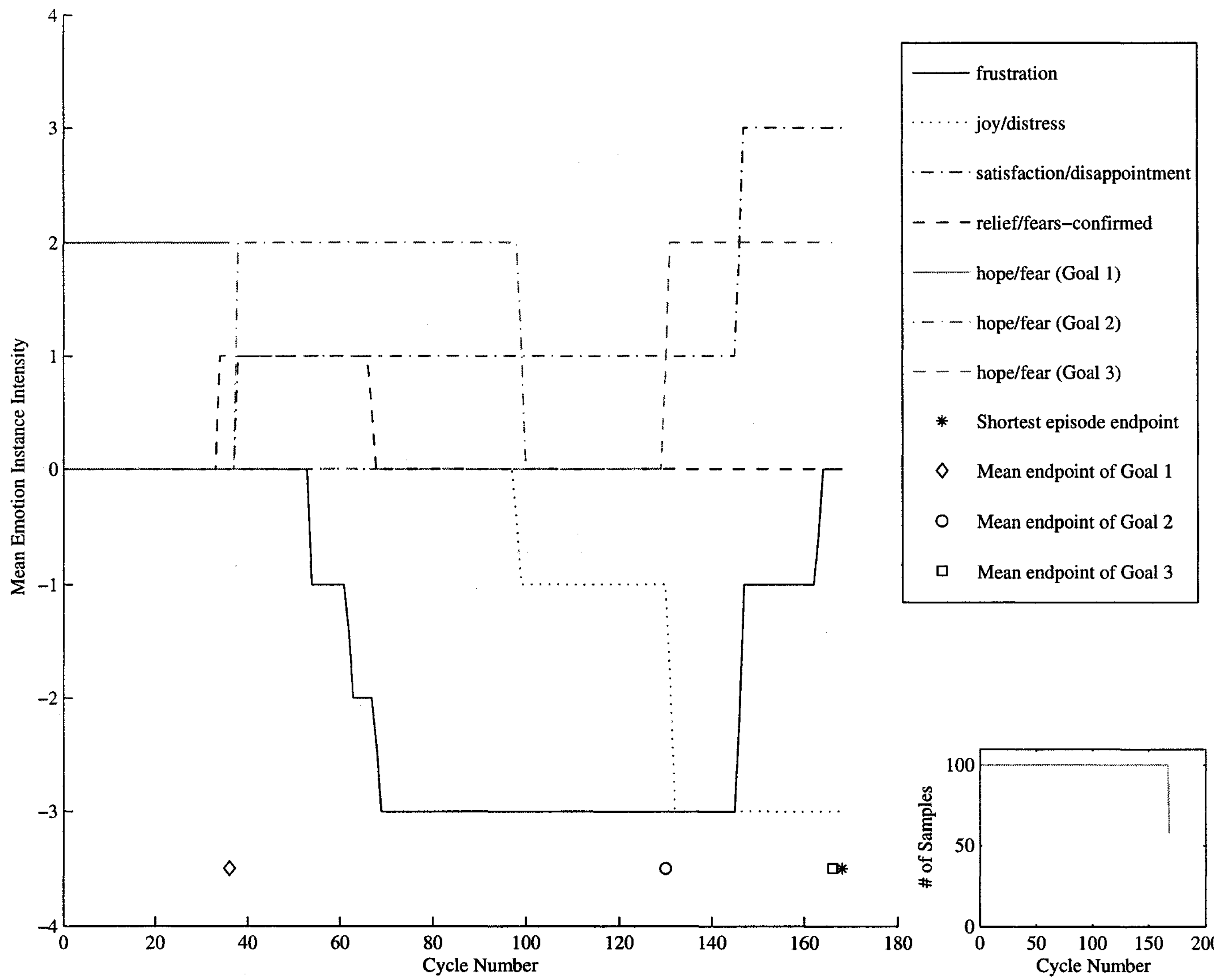

Figure A.3: Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

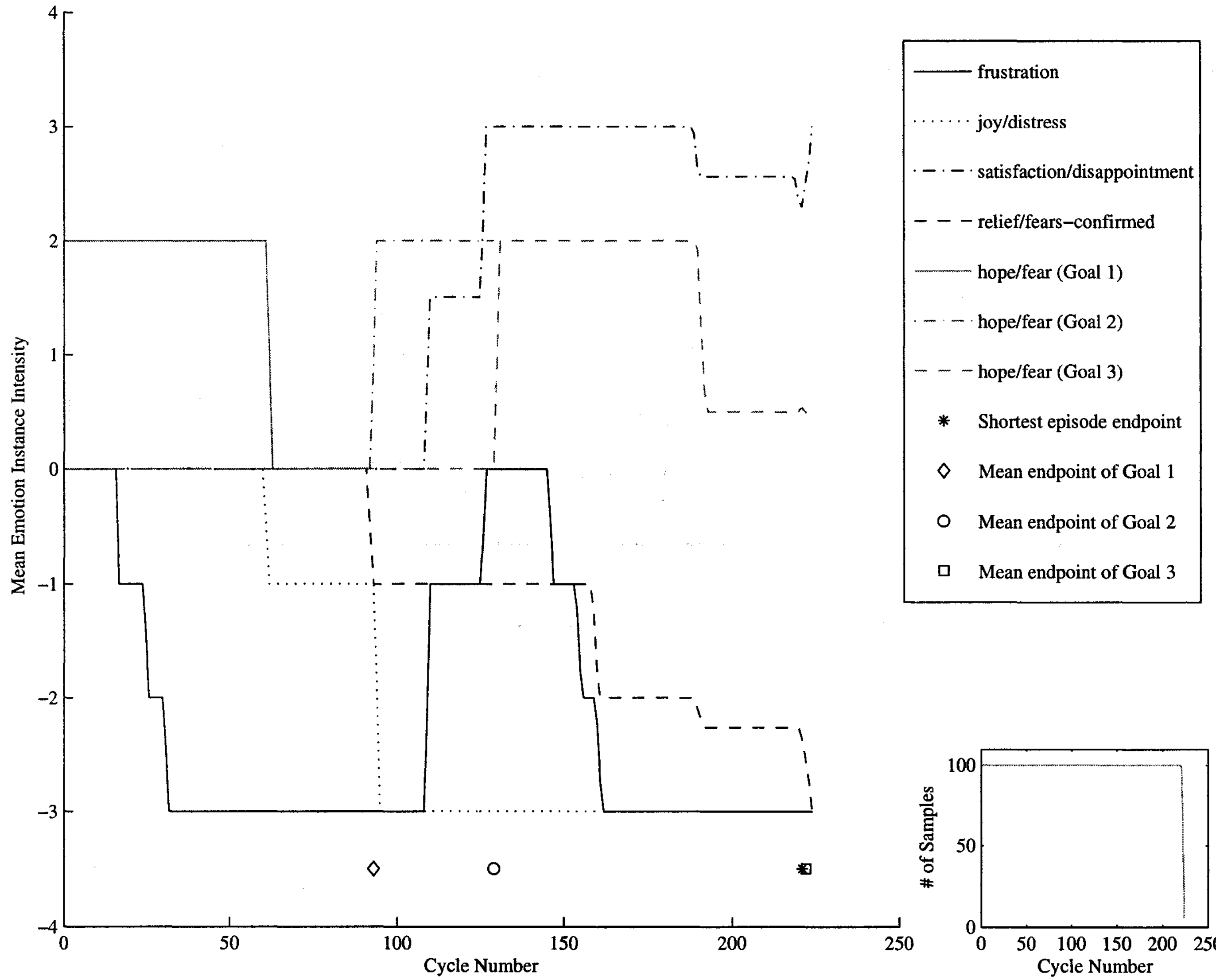

Figure A.4: Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

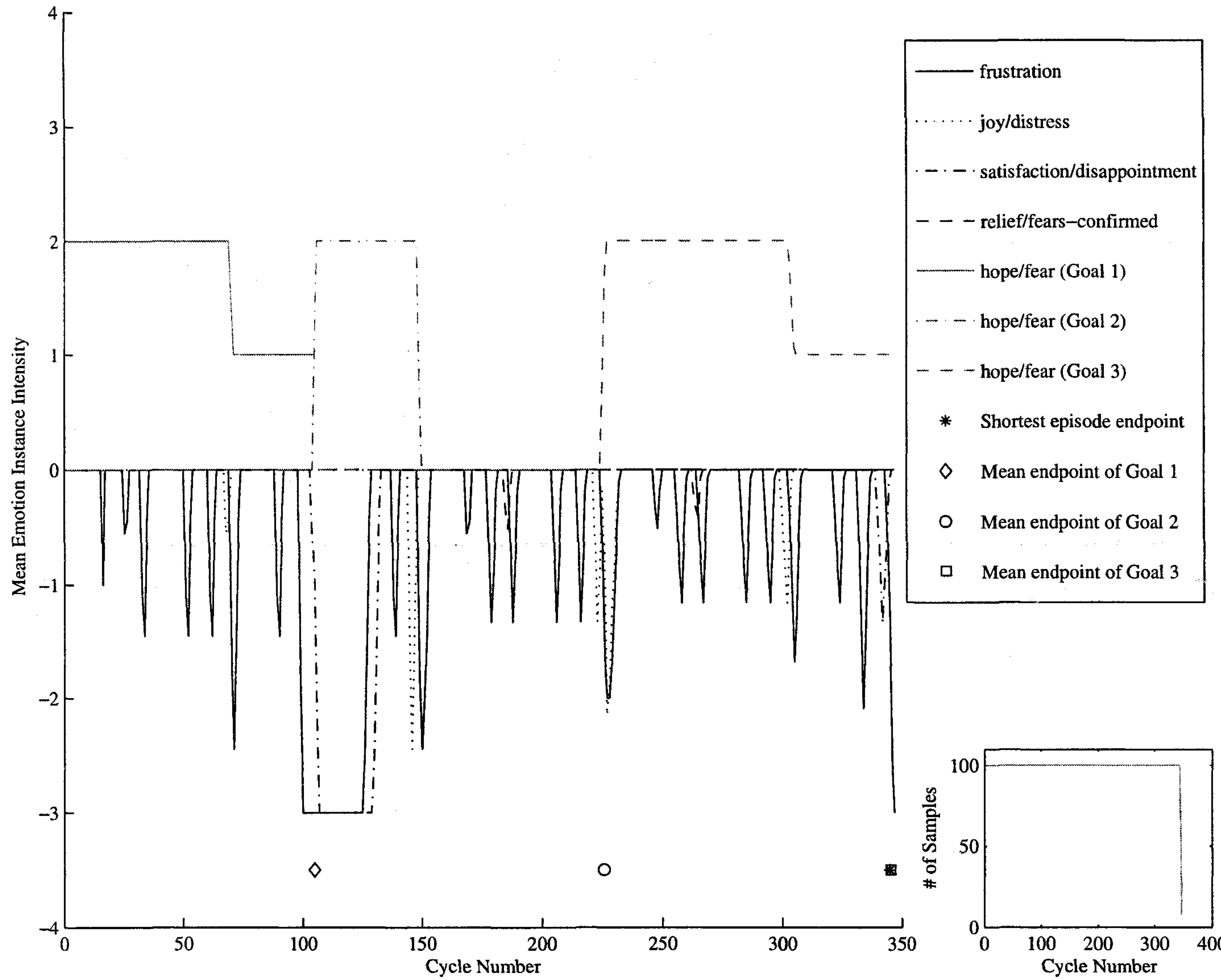

Figure A.5: Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

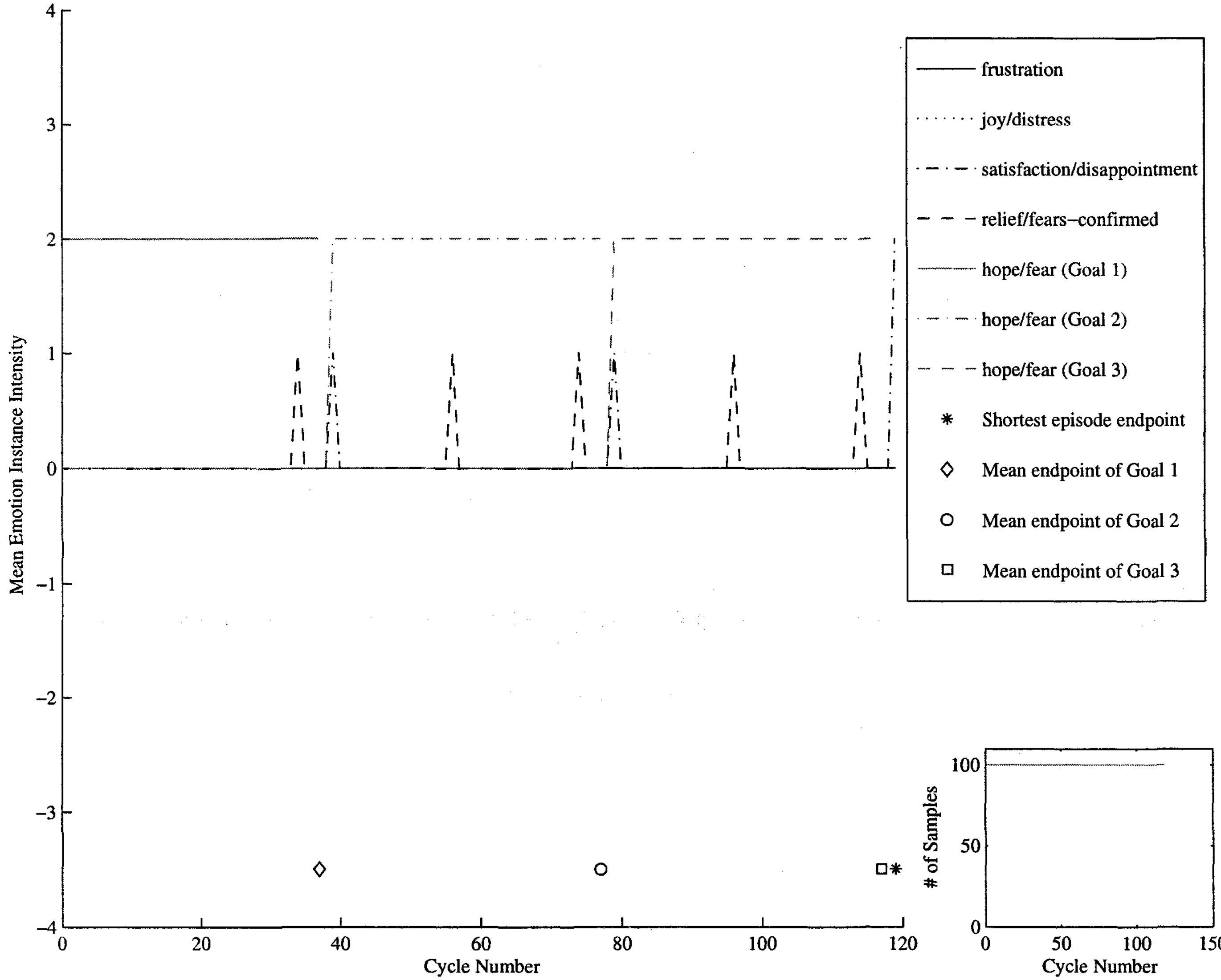

Figure A.6: Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience.

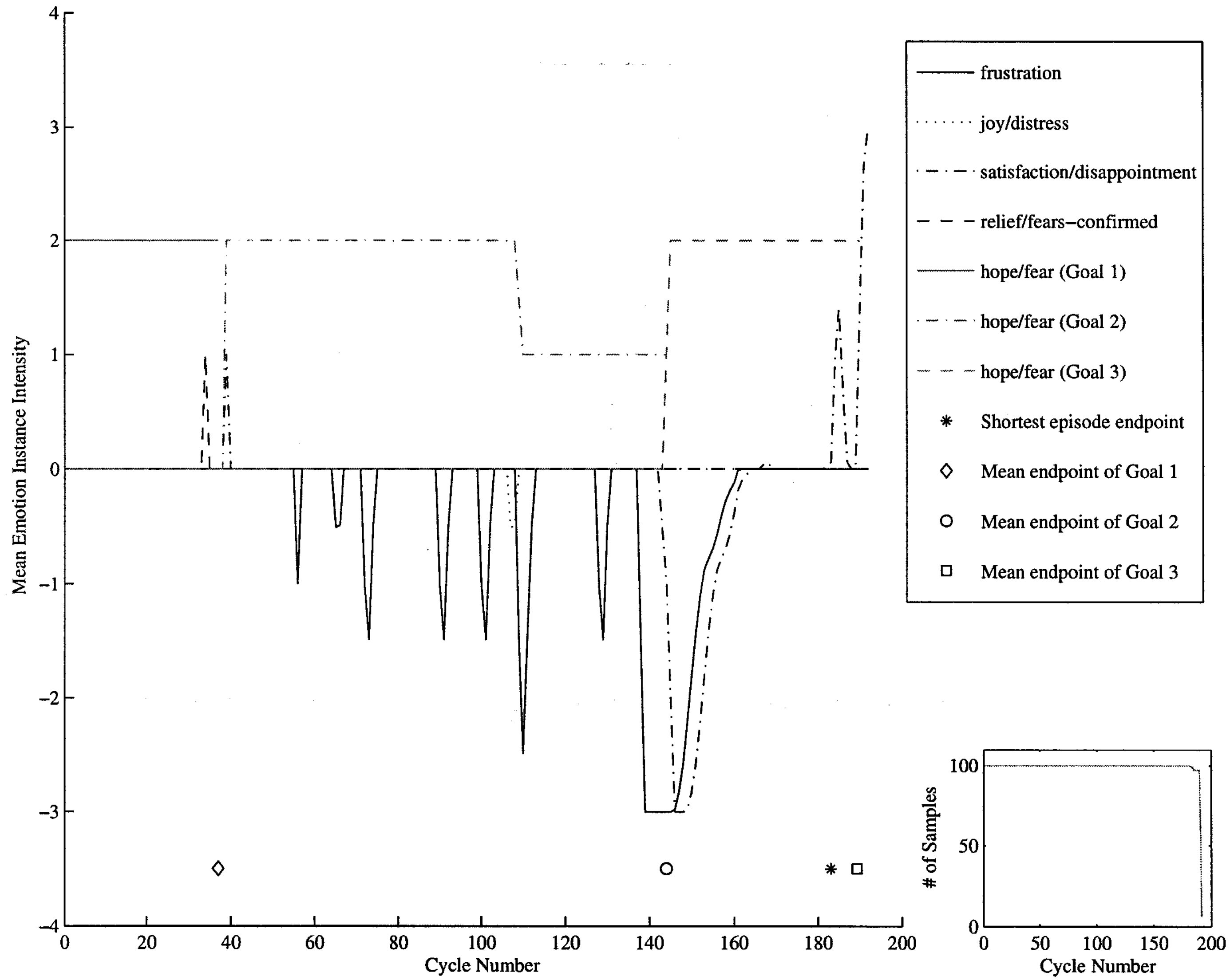

Figure A.7: Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience.

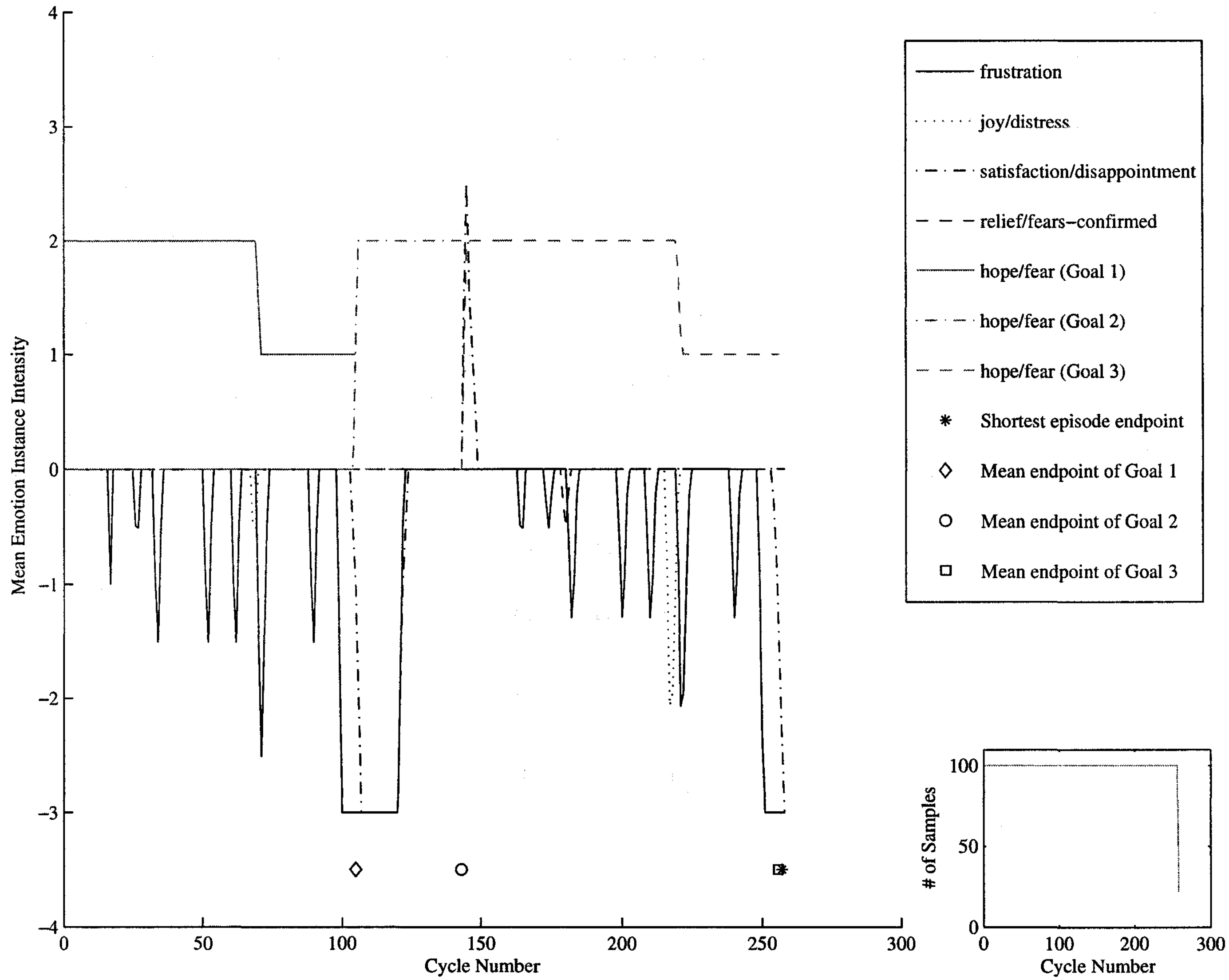

Figure A.8: Emotion chart for the non-noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

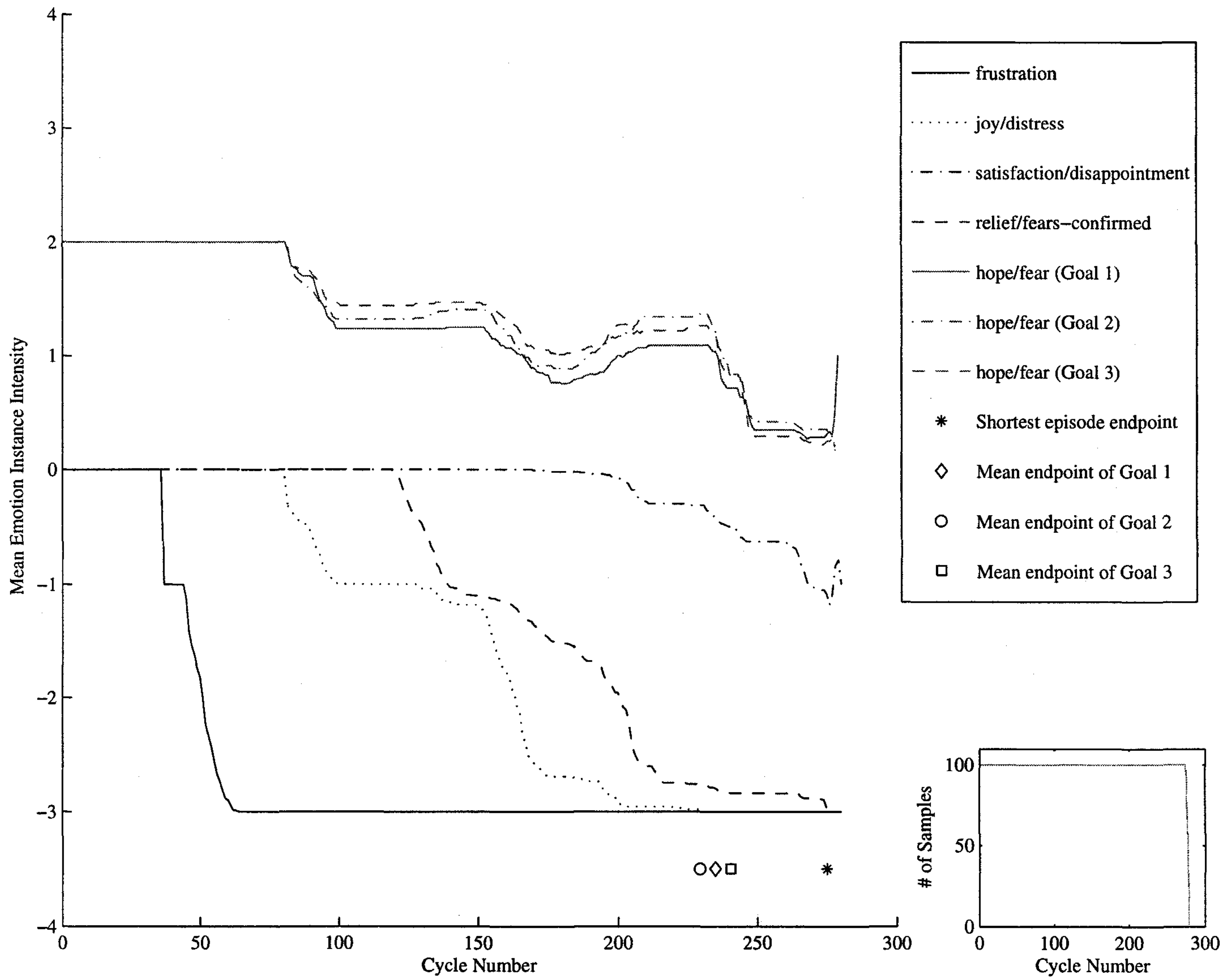

Figure A.9: Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

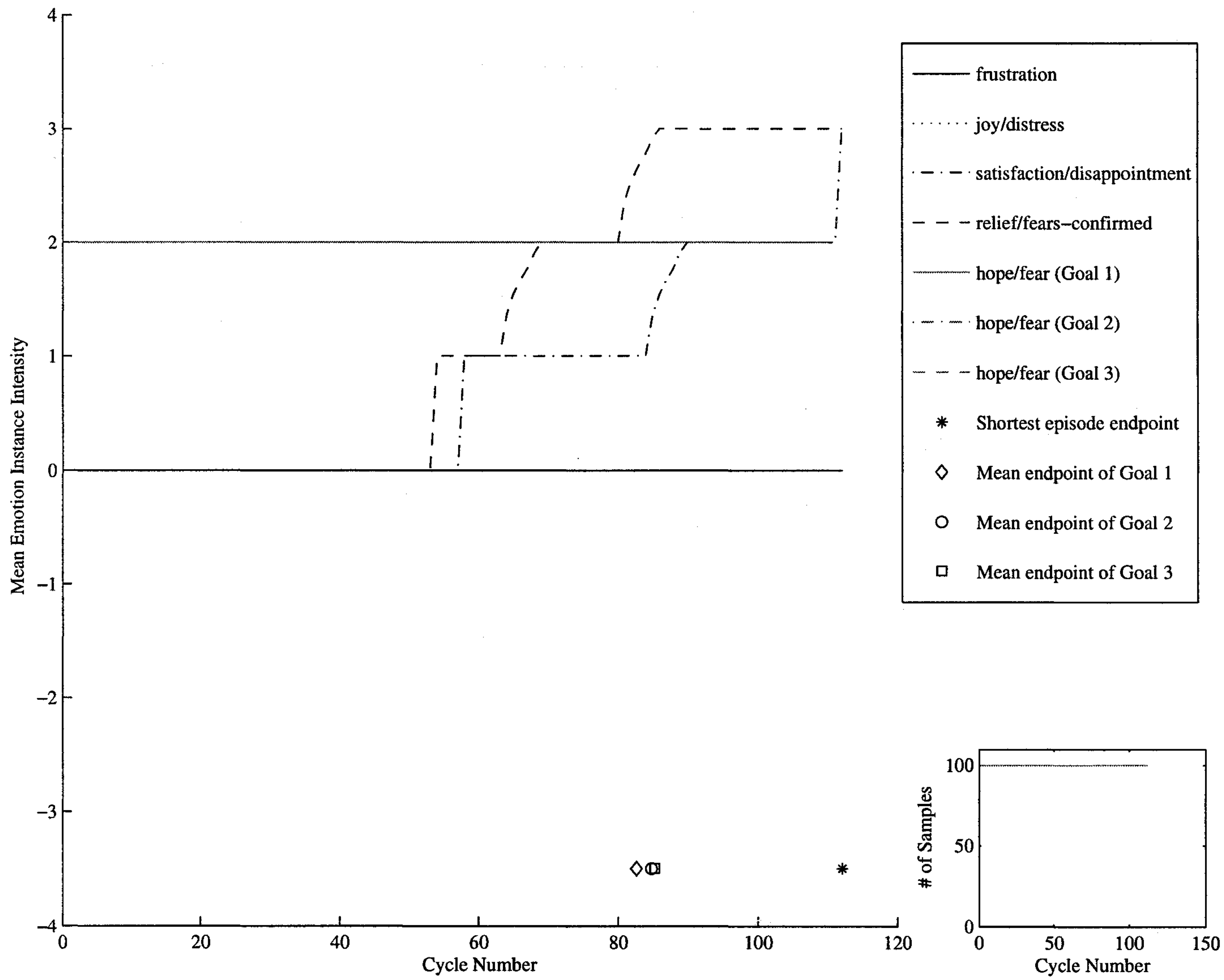

Figure A.10: Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience.

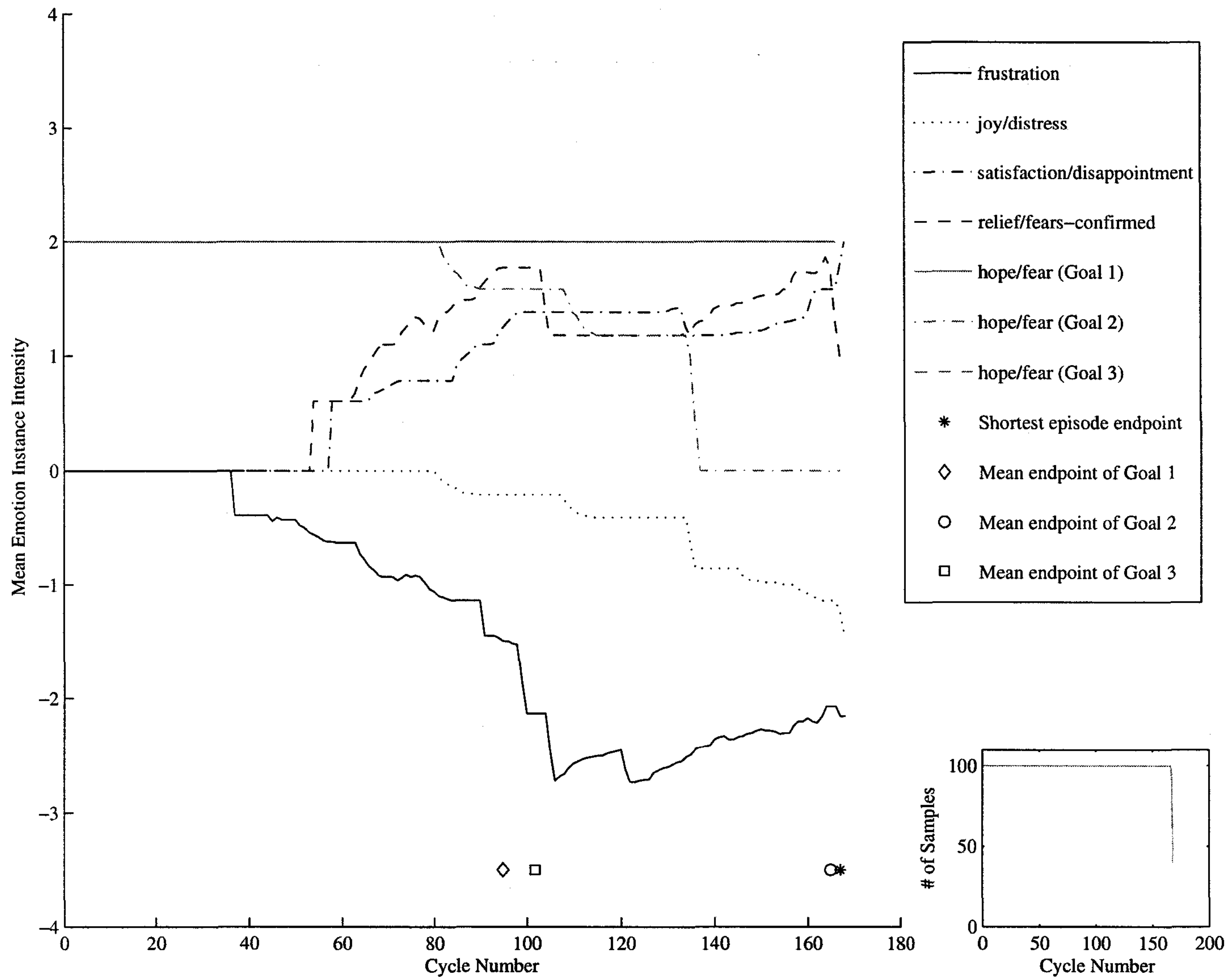

Figure A.11: Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

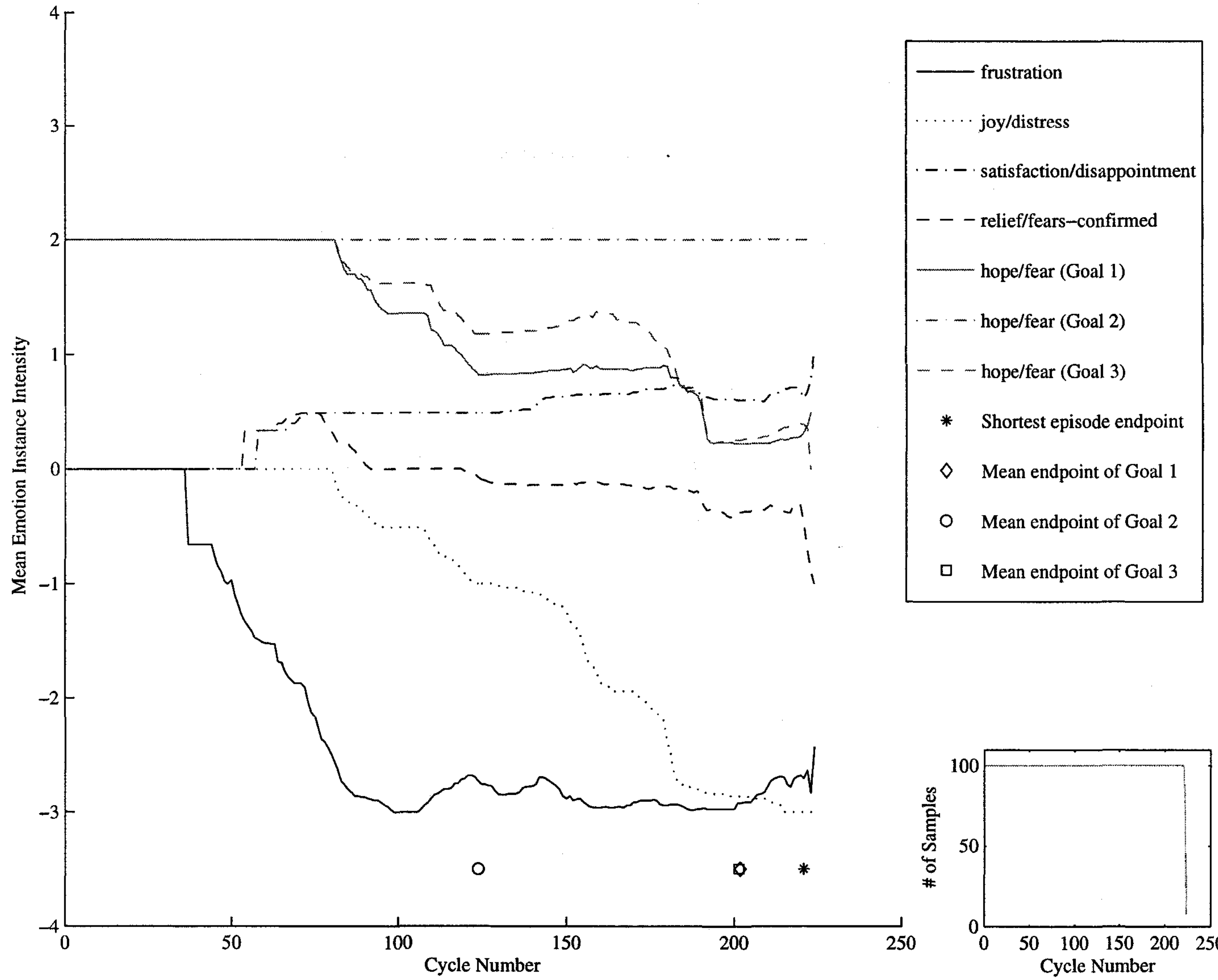

Figure A.12: Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

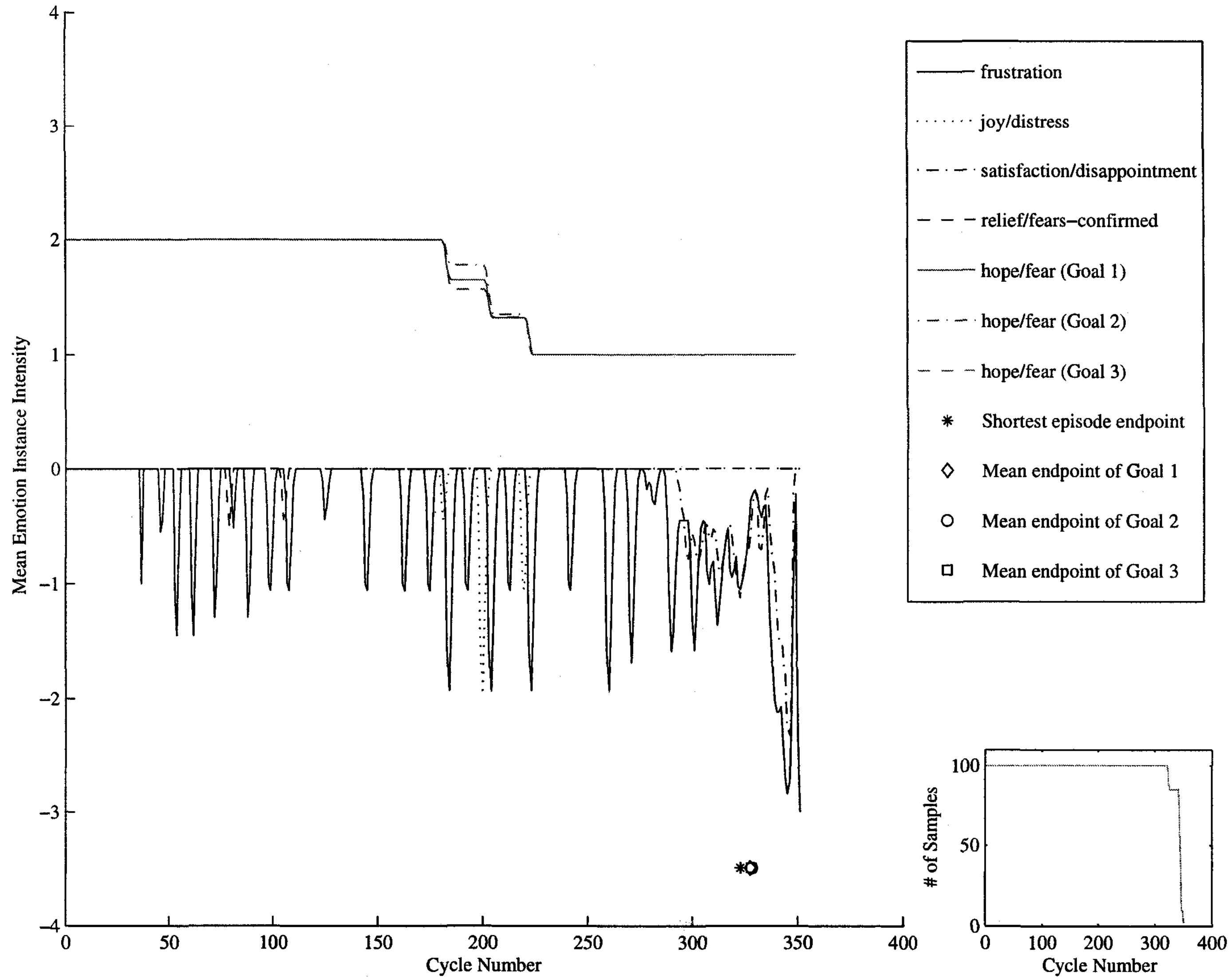

Figure A.13: Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

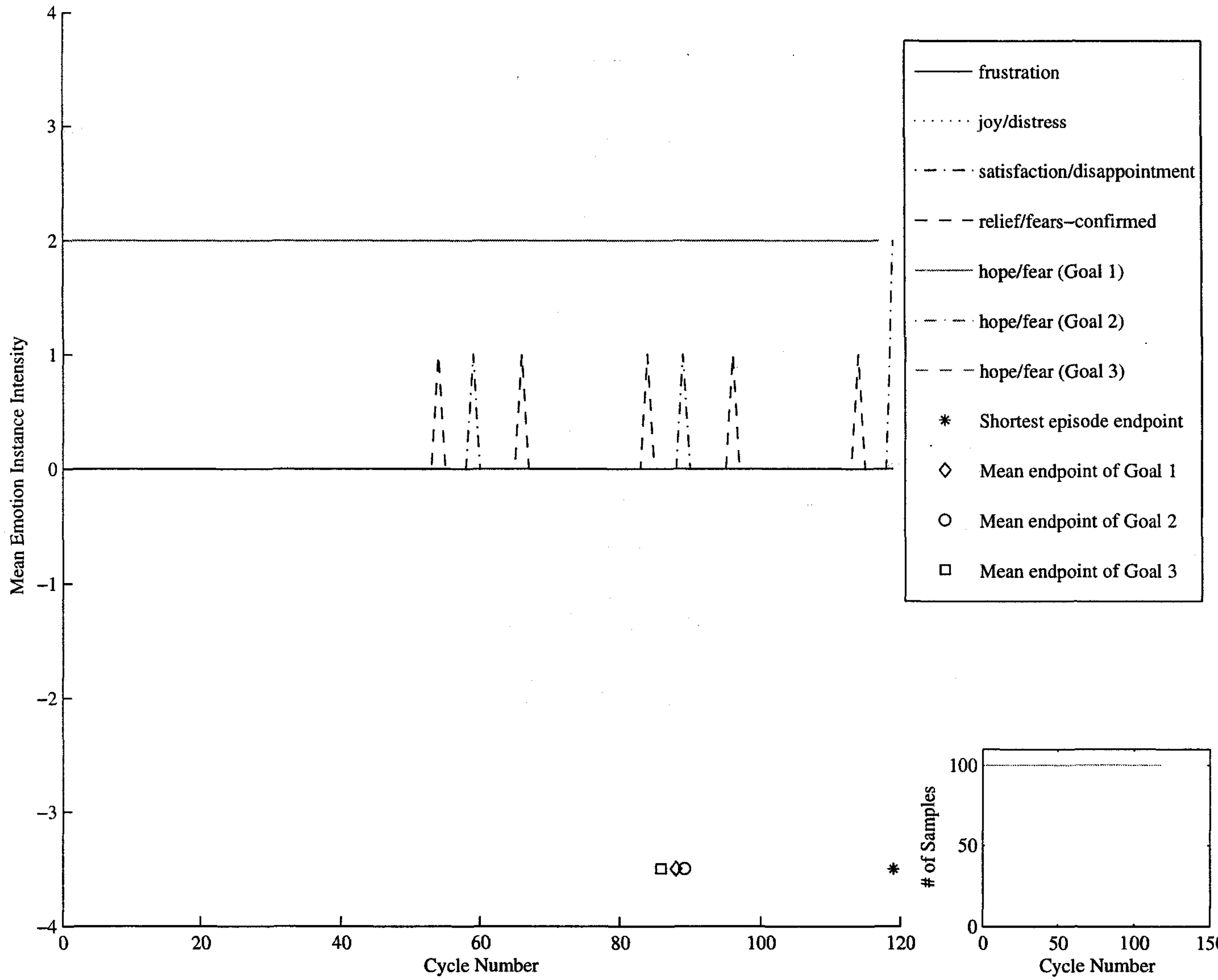

Figure A.14: Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience.

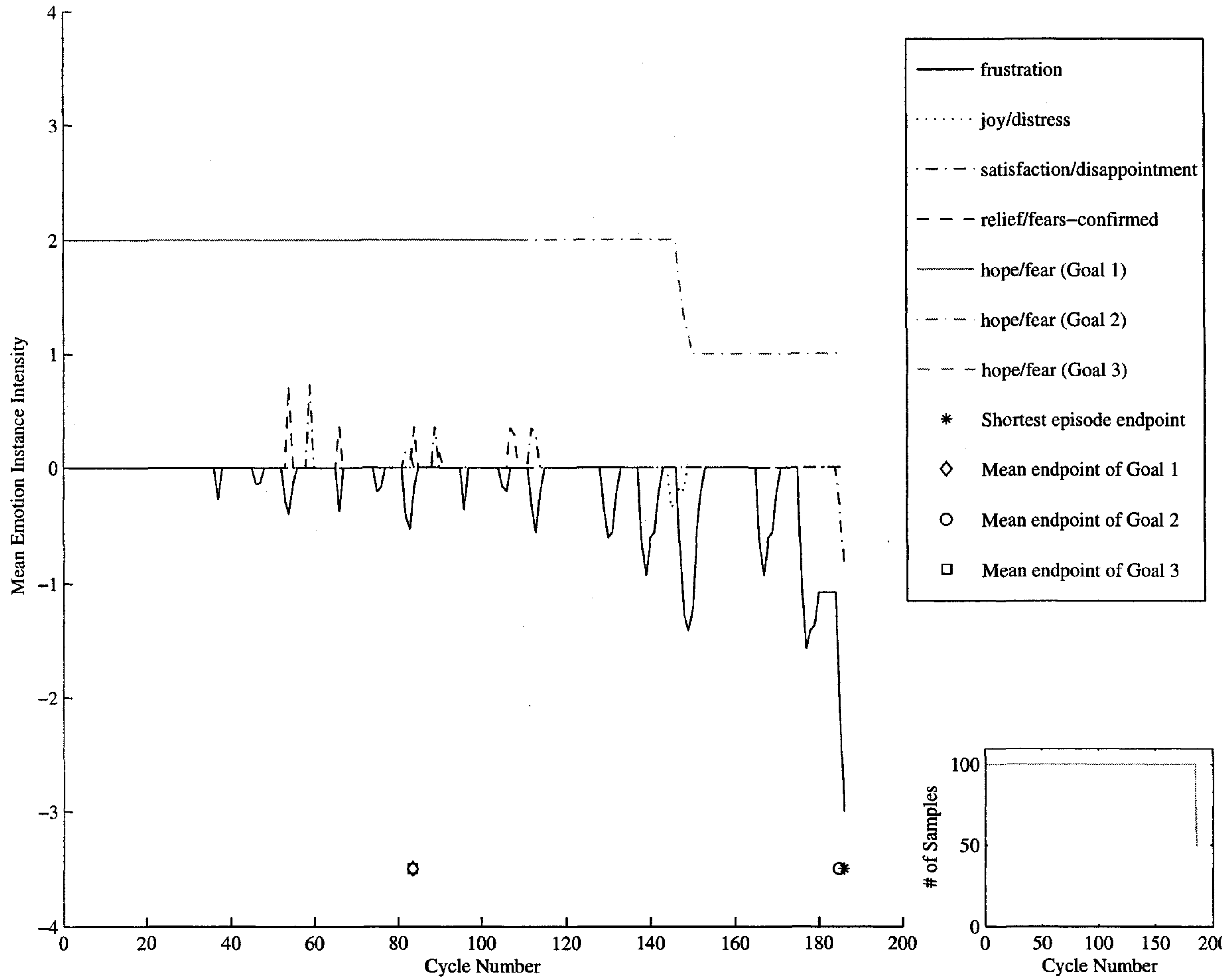

Figure A.15: Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience.

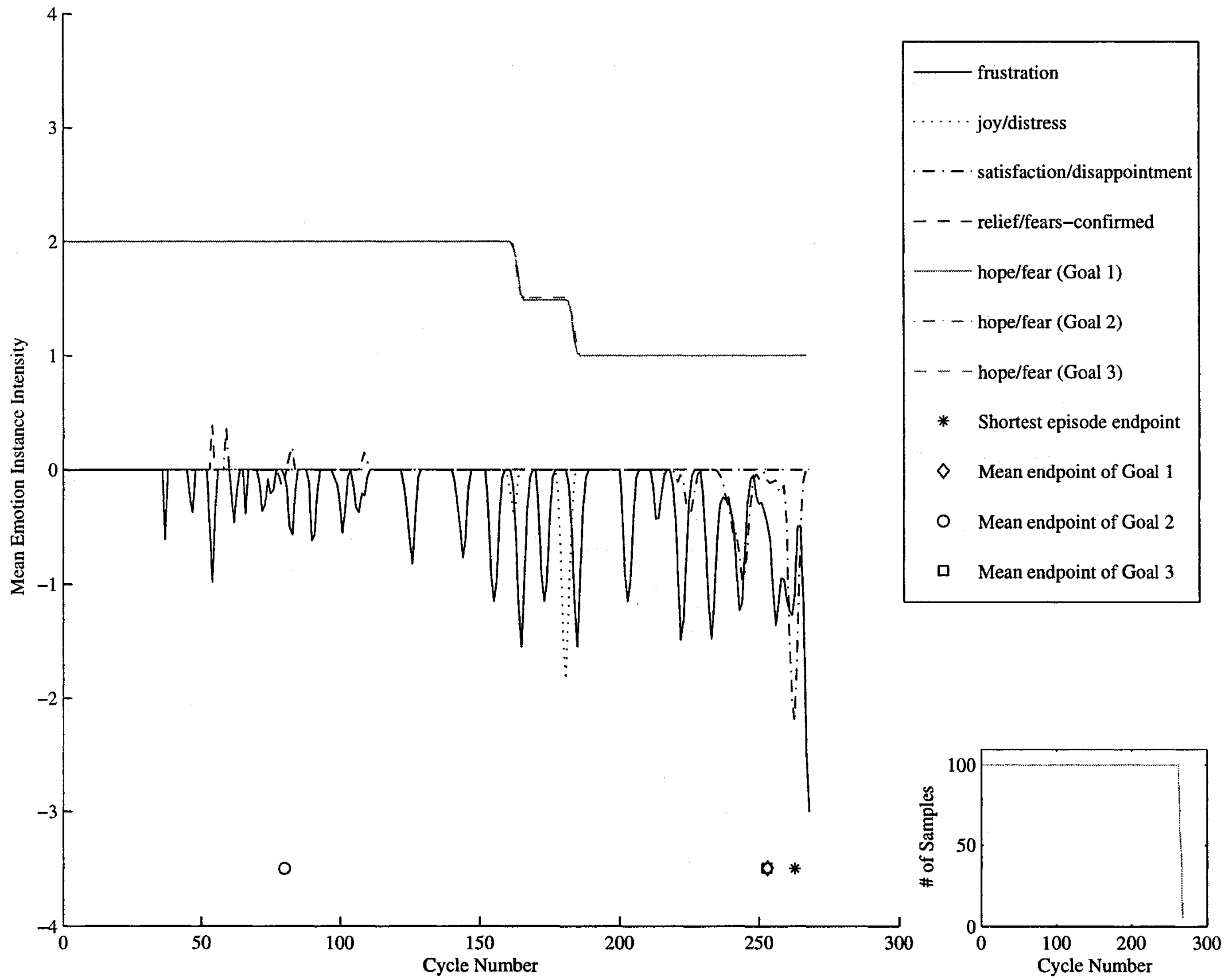

Figure A.16: Emotion chart for the non-noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

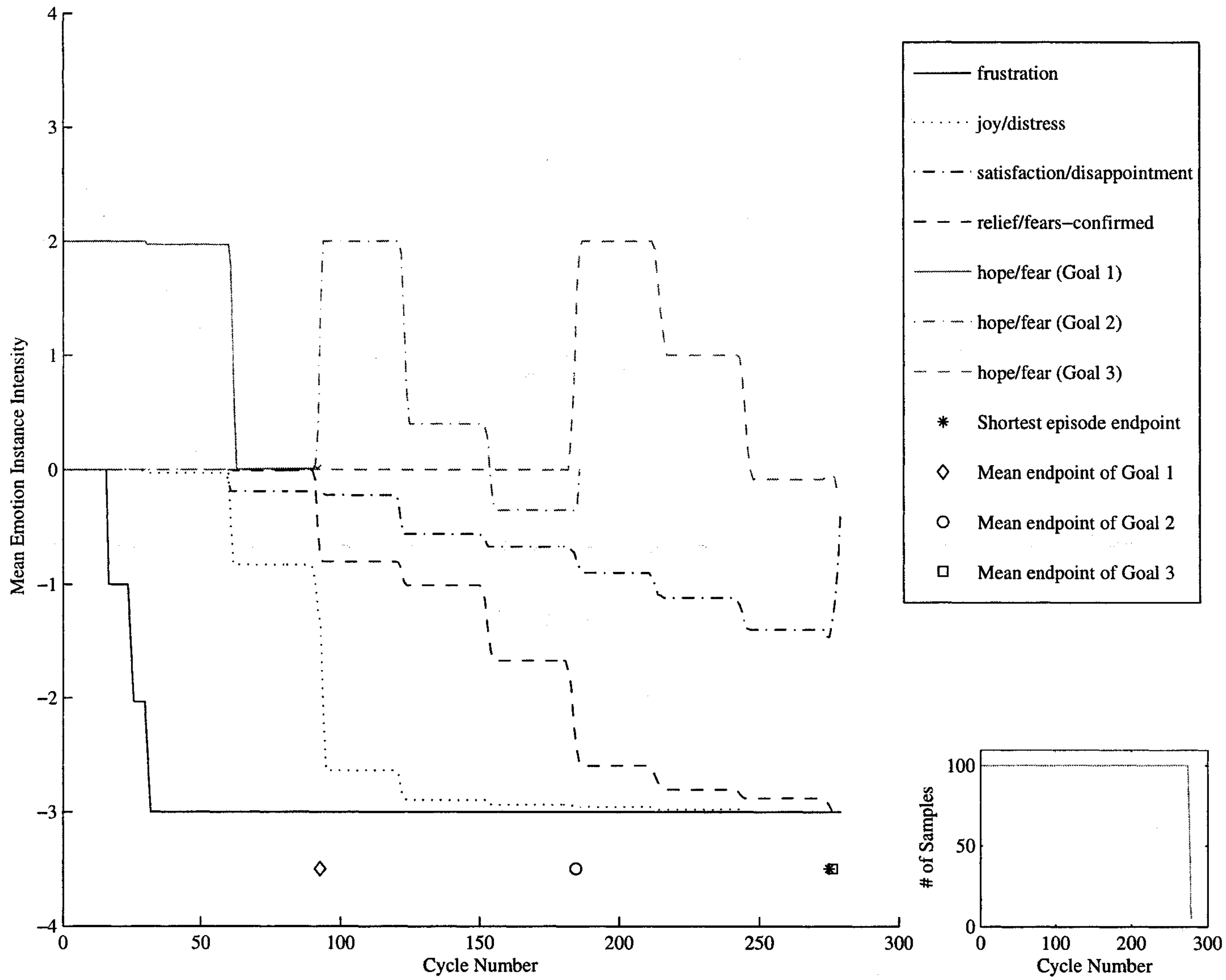

Figure A.17: Emotion chart for the noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

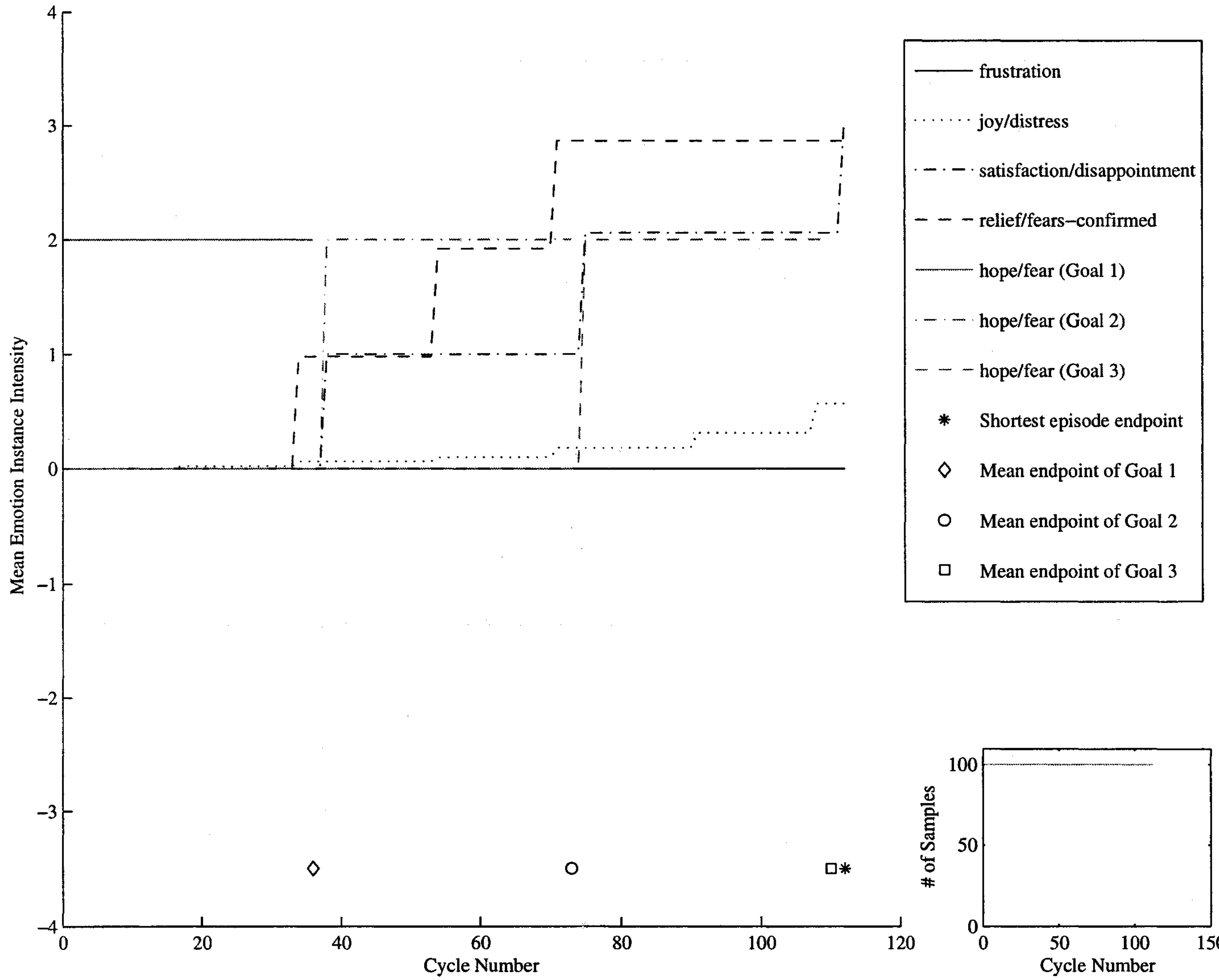

Figure A.18: Emotion chart for the noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience.

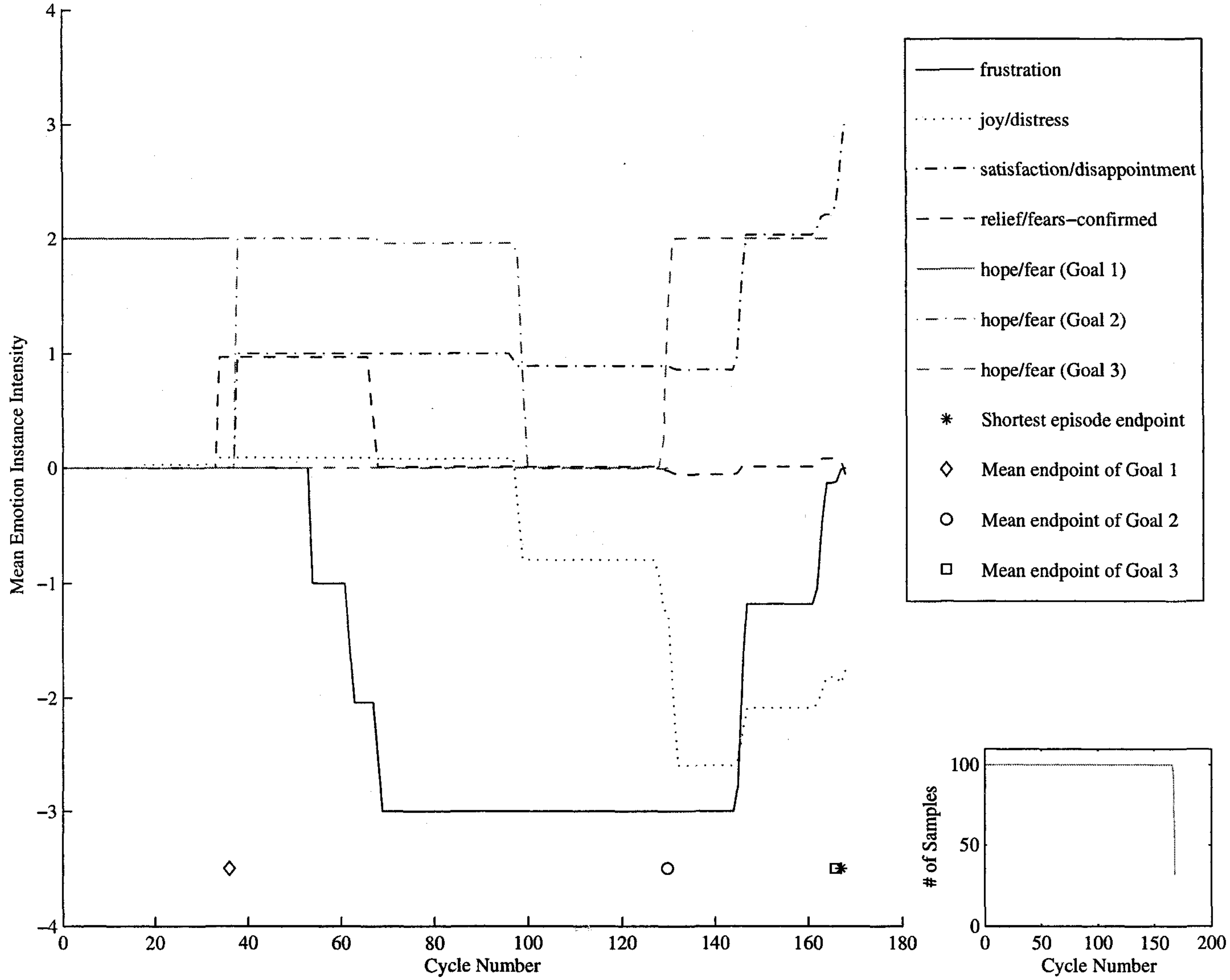

Figure A.19: Emotion chart for the noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

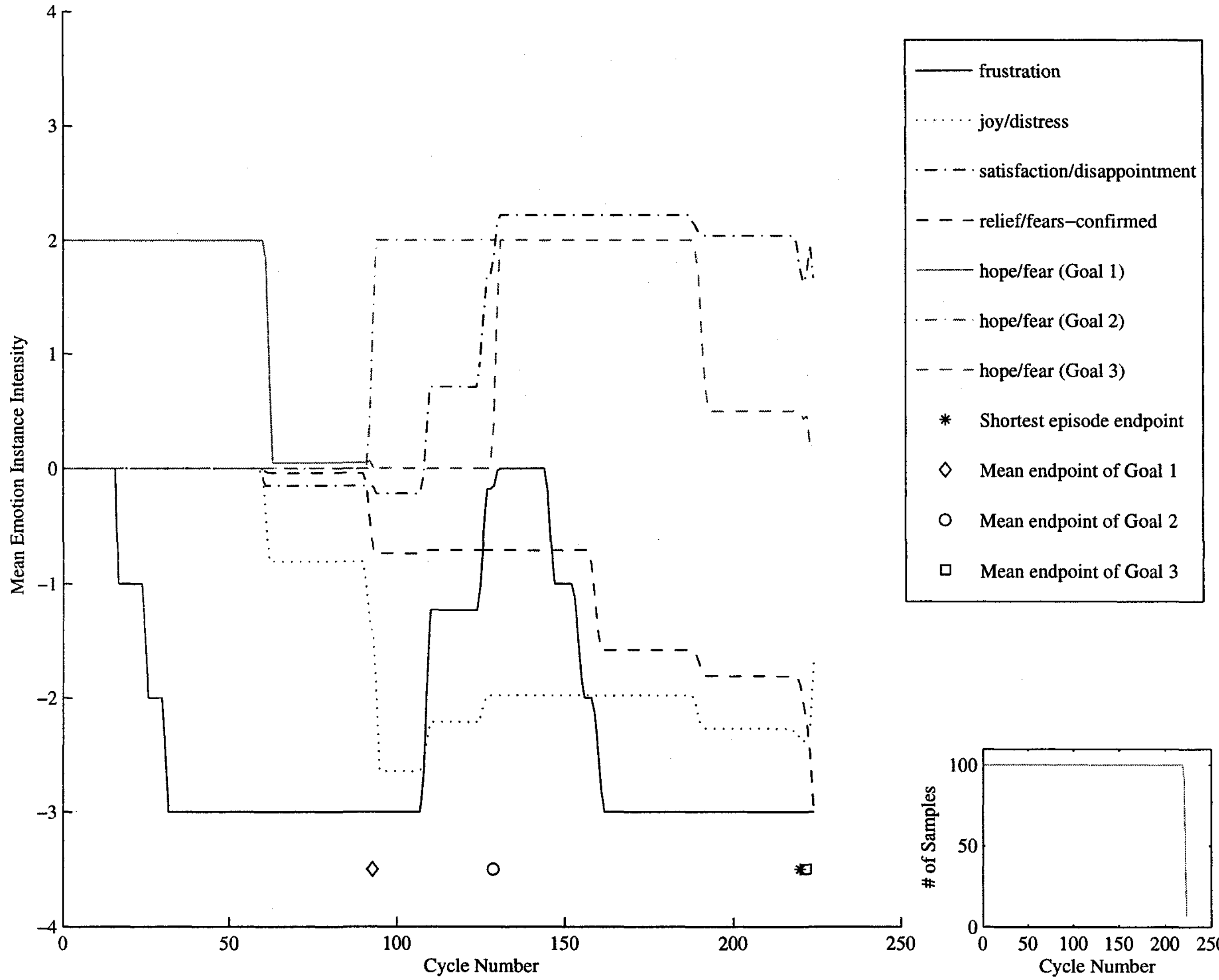

Figure A.20: Emotion chart for the noisy emotional agent in the symmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

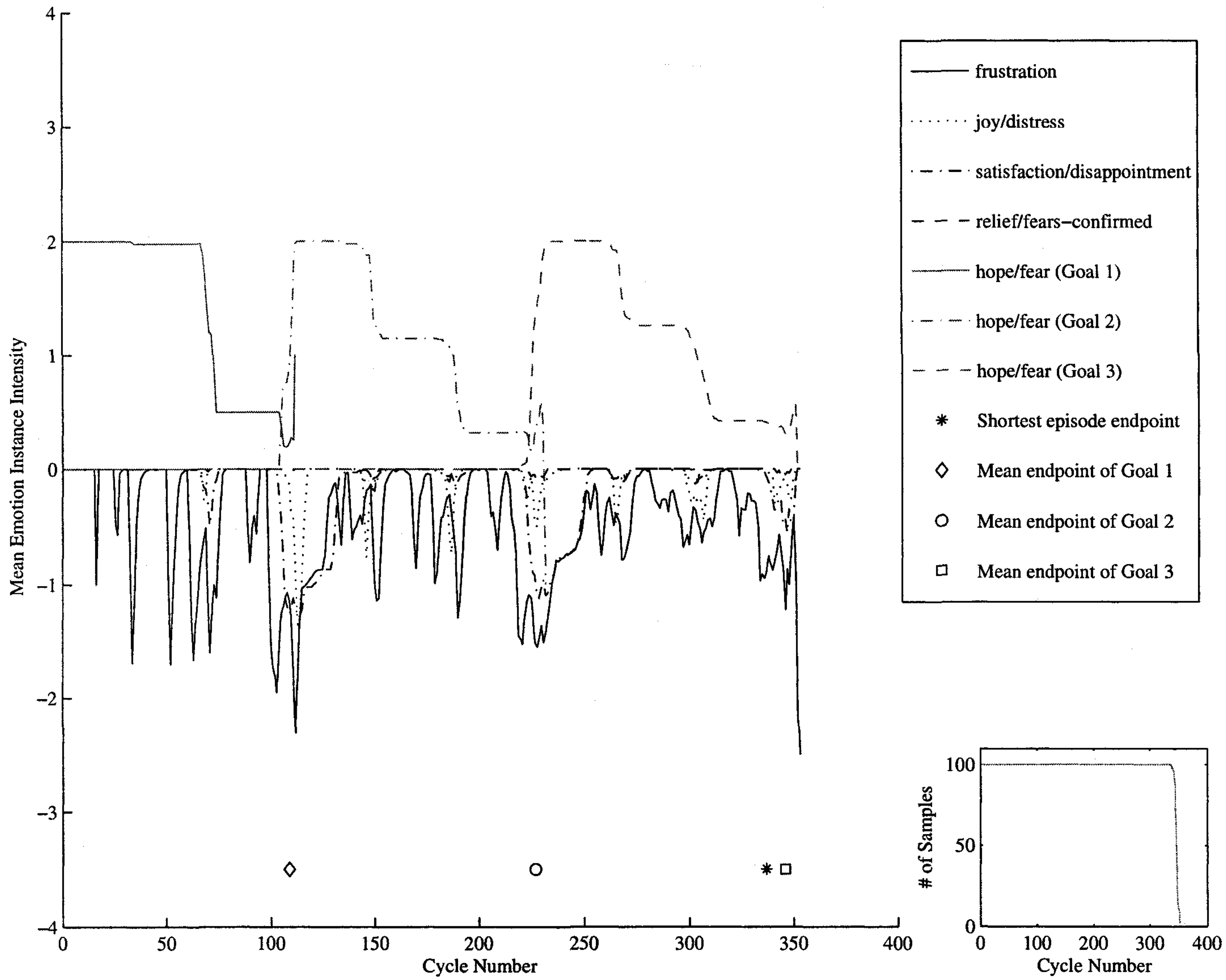

Figure A.21: Emotion chart for the noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

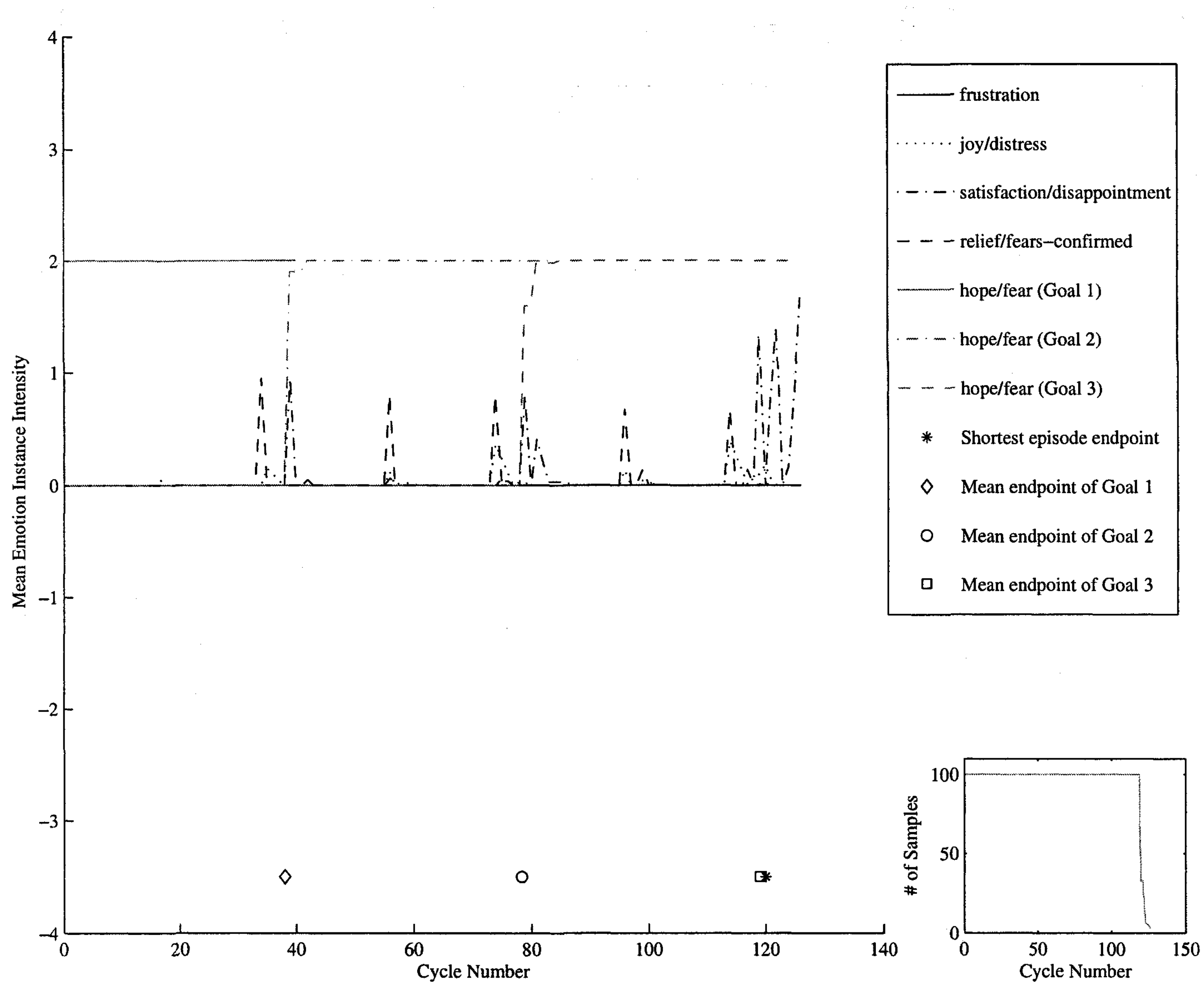

Figure A.22: Emotion chart for the noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience.

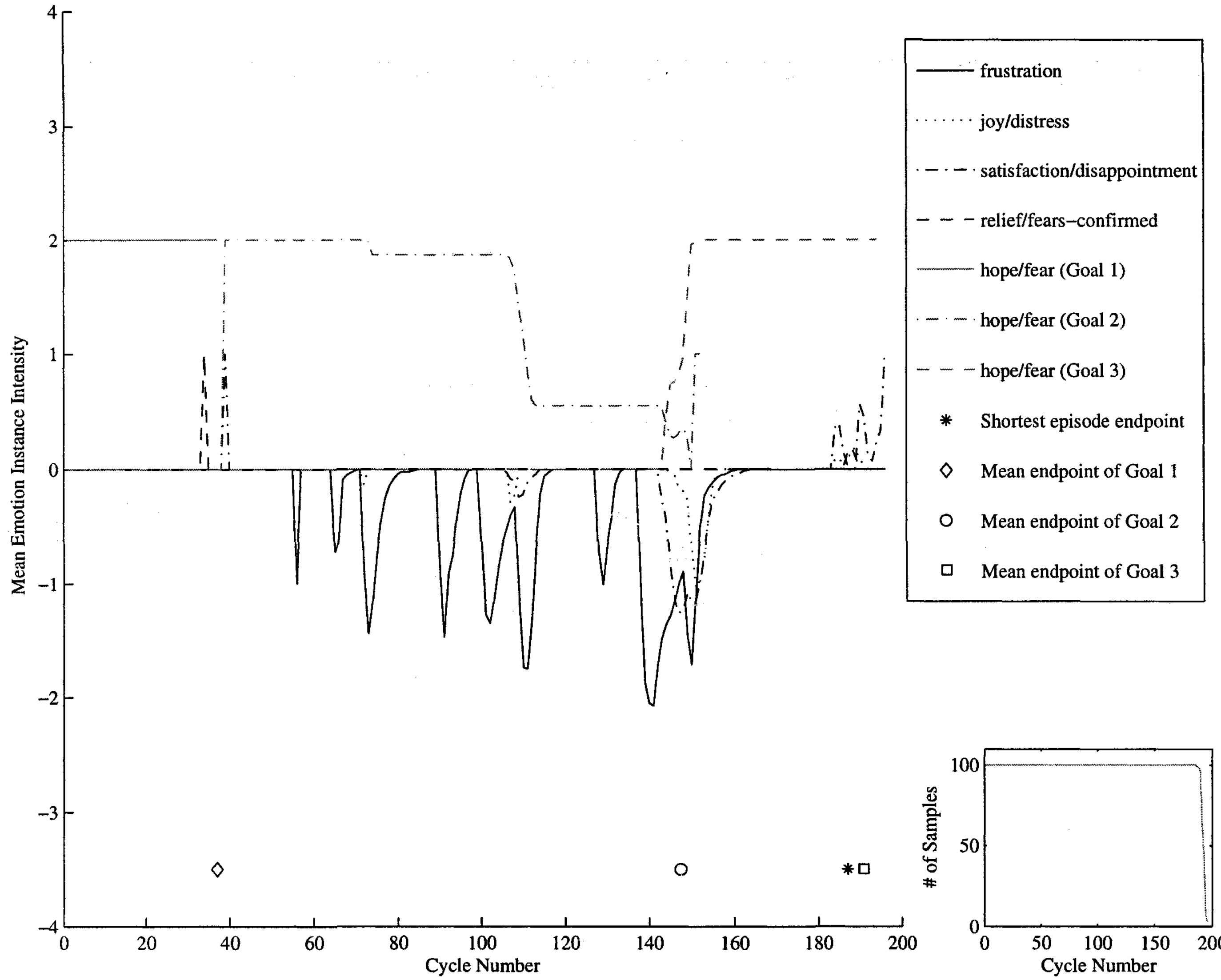

Figure A.23: Emotion chart for the noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience.

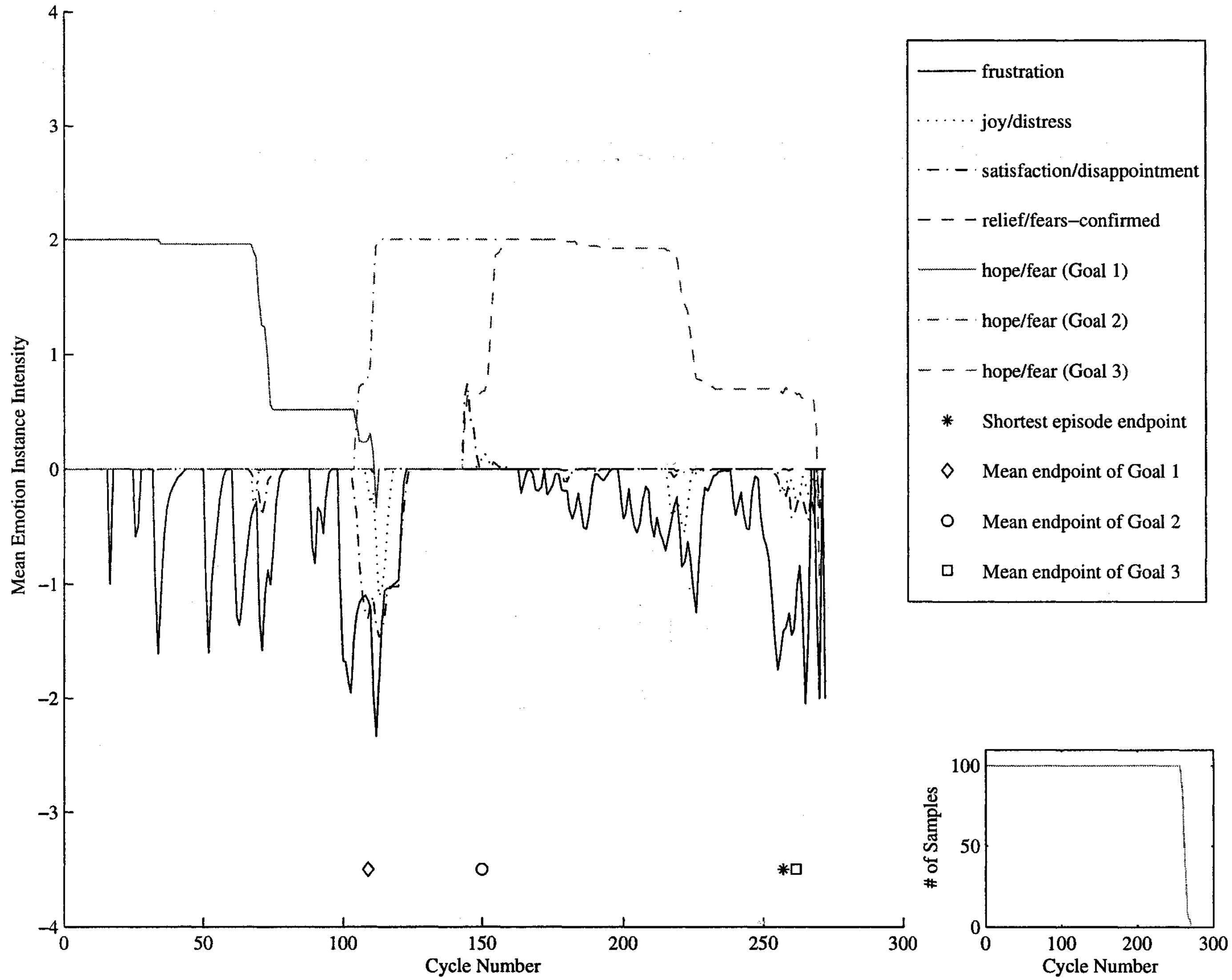

Figure A.24: Emotion chart for the noisy emotional agent in the symmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

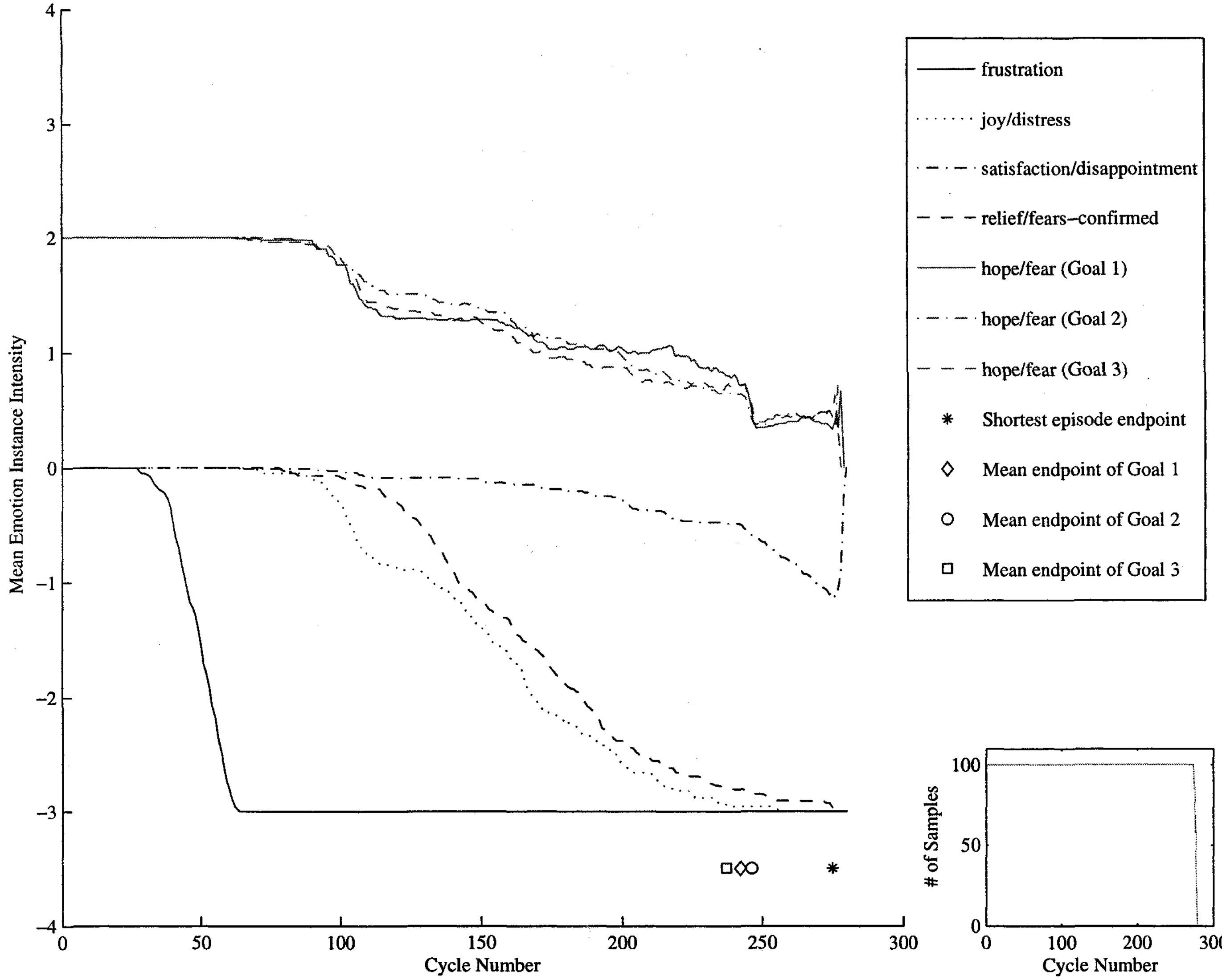

Figure A.25: Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

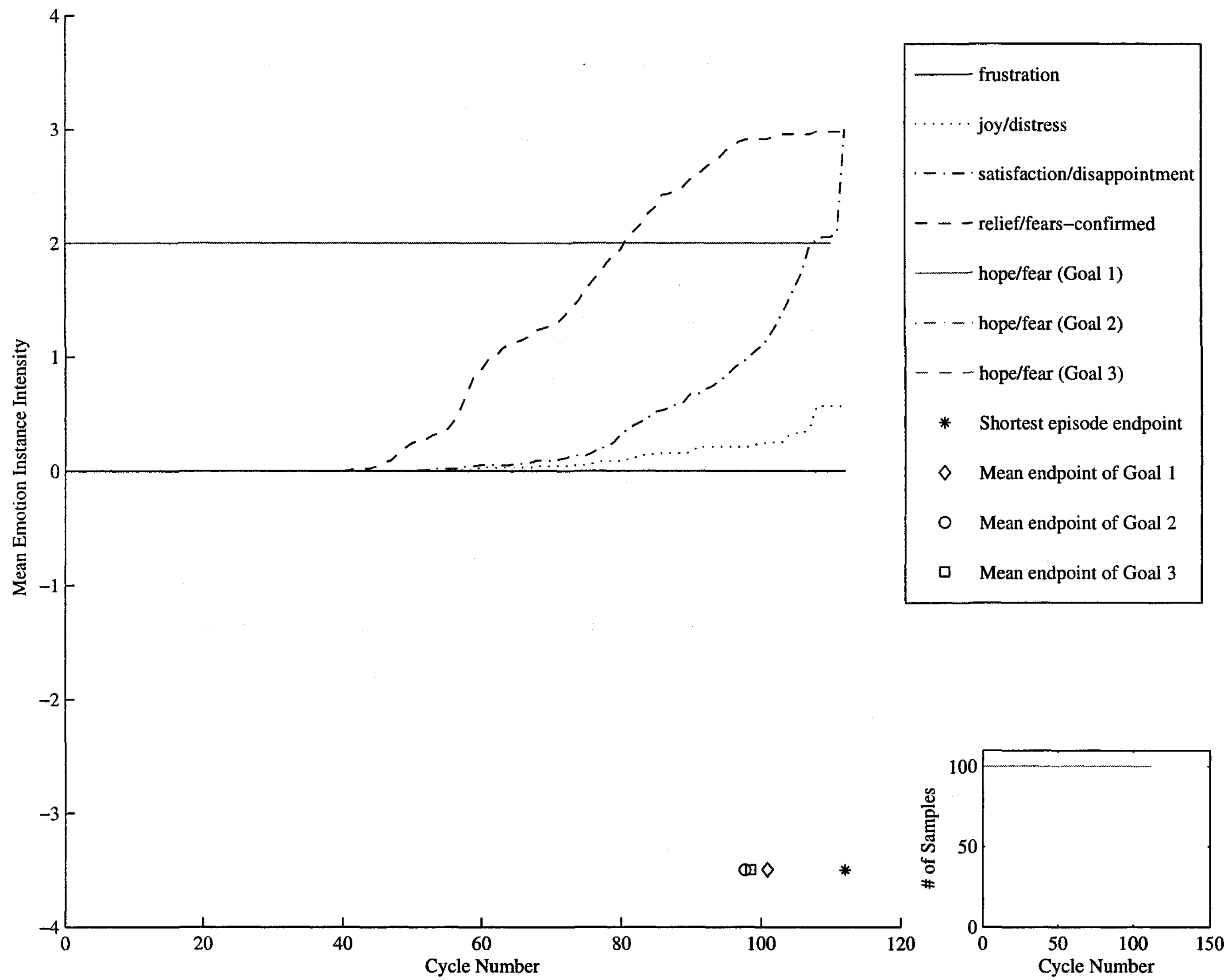

Figure A.26: Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience.

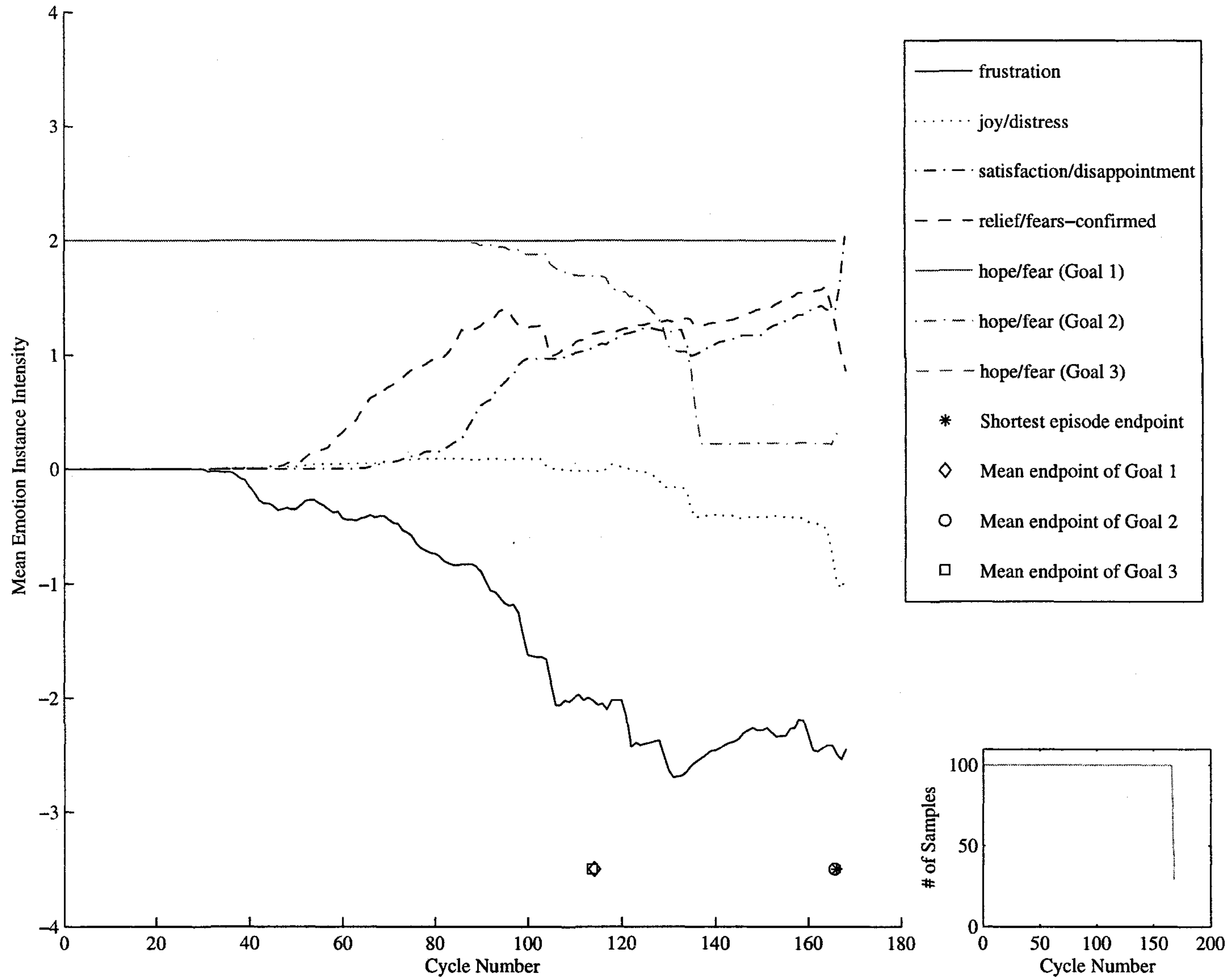

Figure A.27: Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

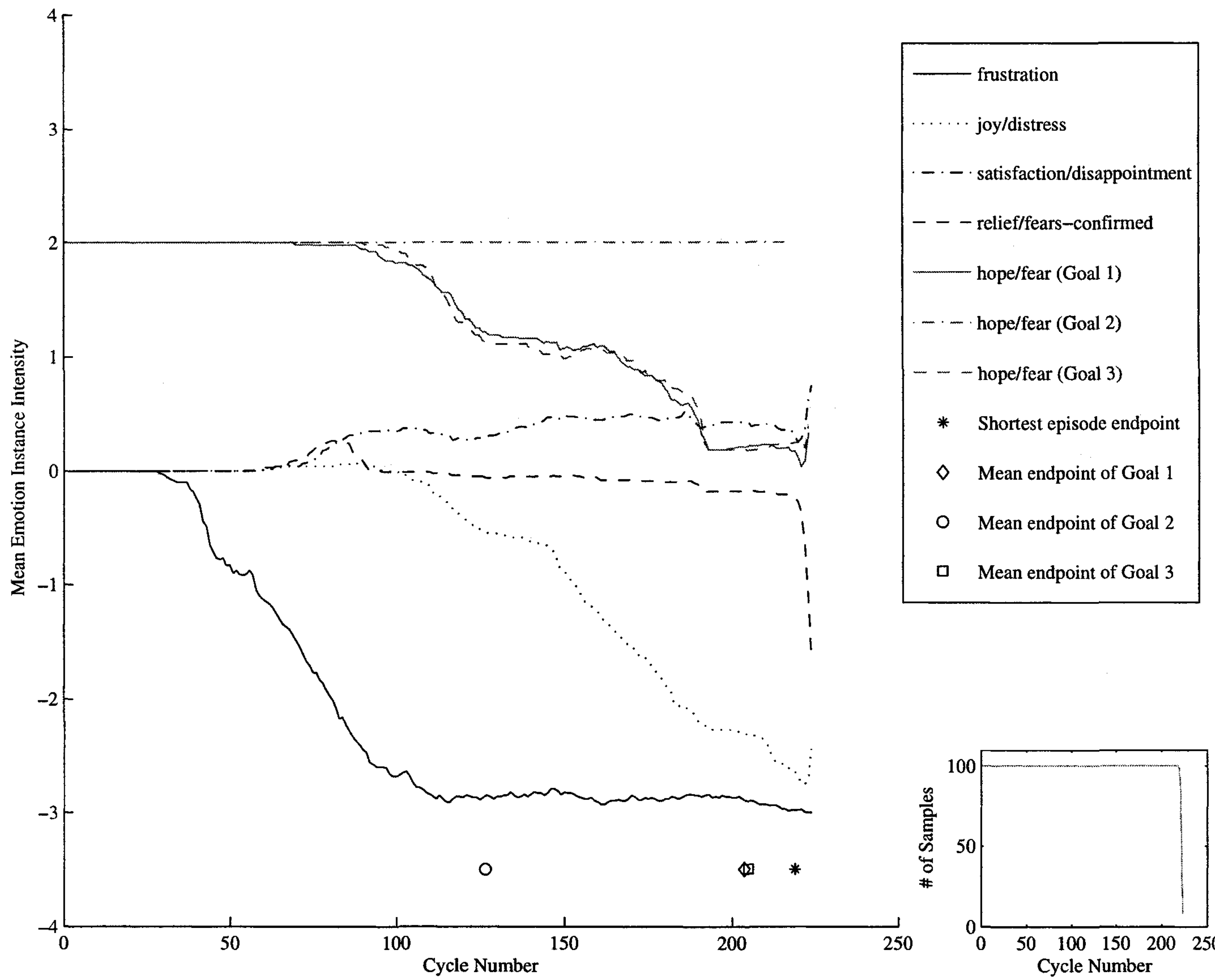

Figure A.28: Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

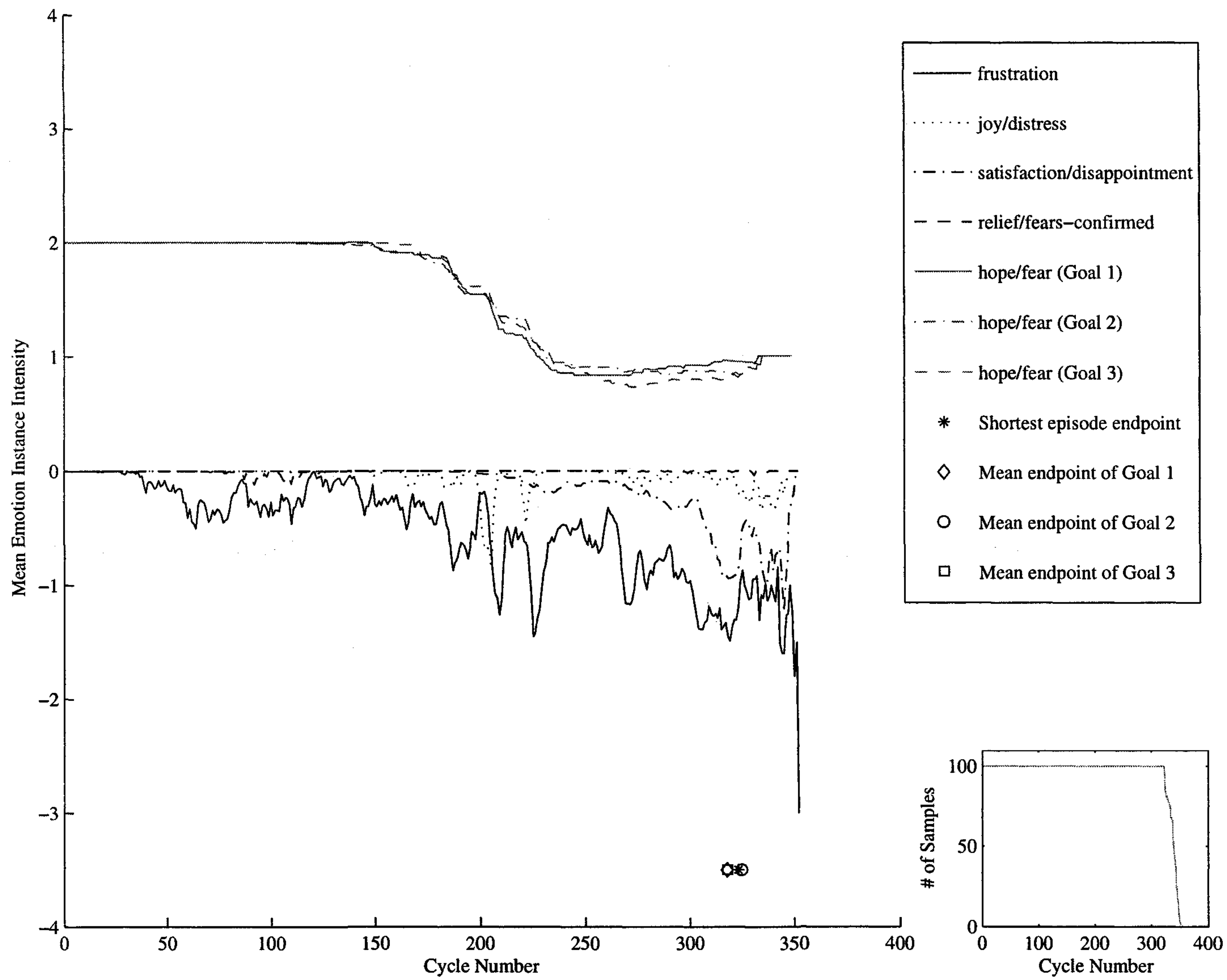

Figure A.29: Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

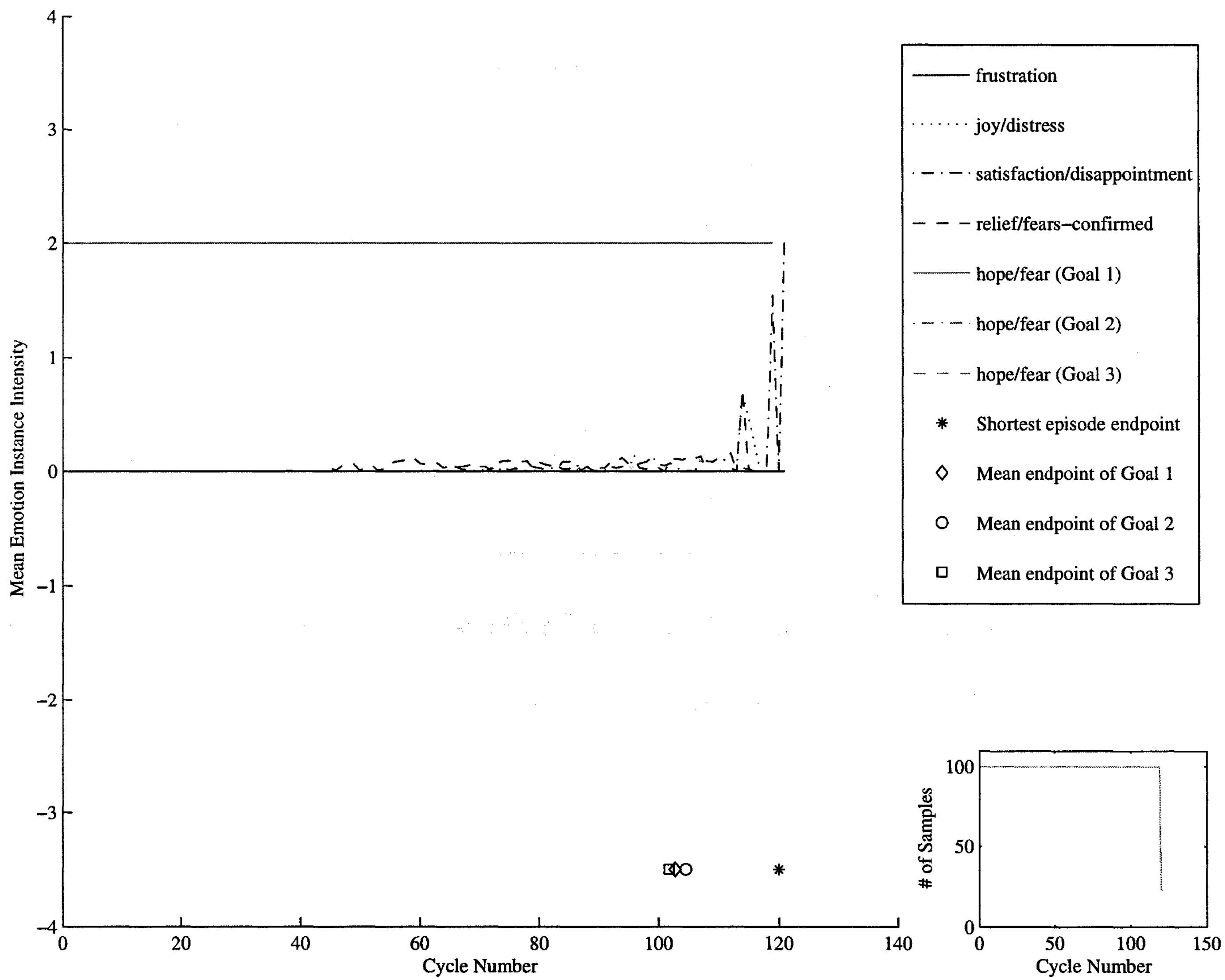

Figure A.30: Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience.

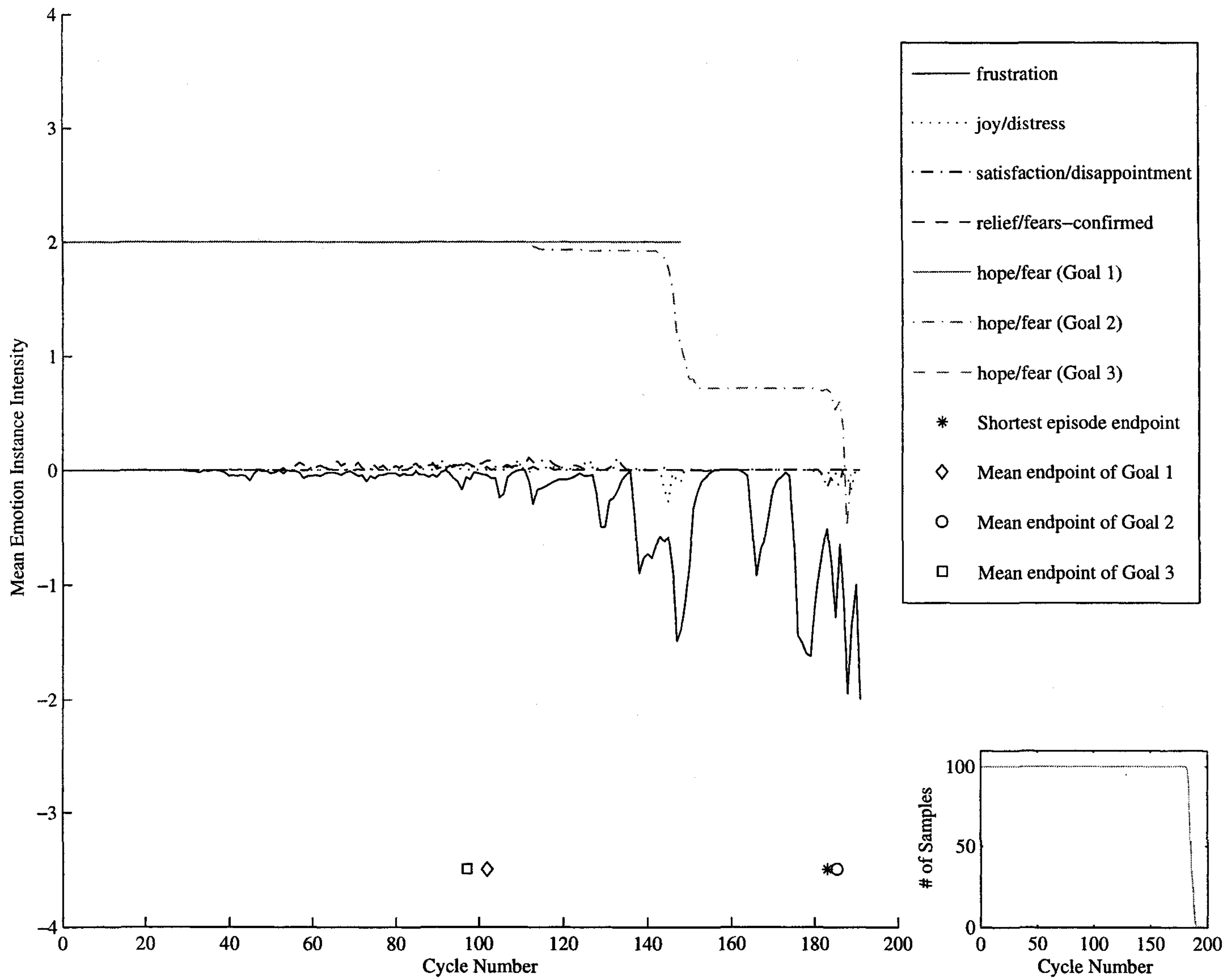

Figure A.31: Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience.

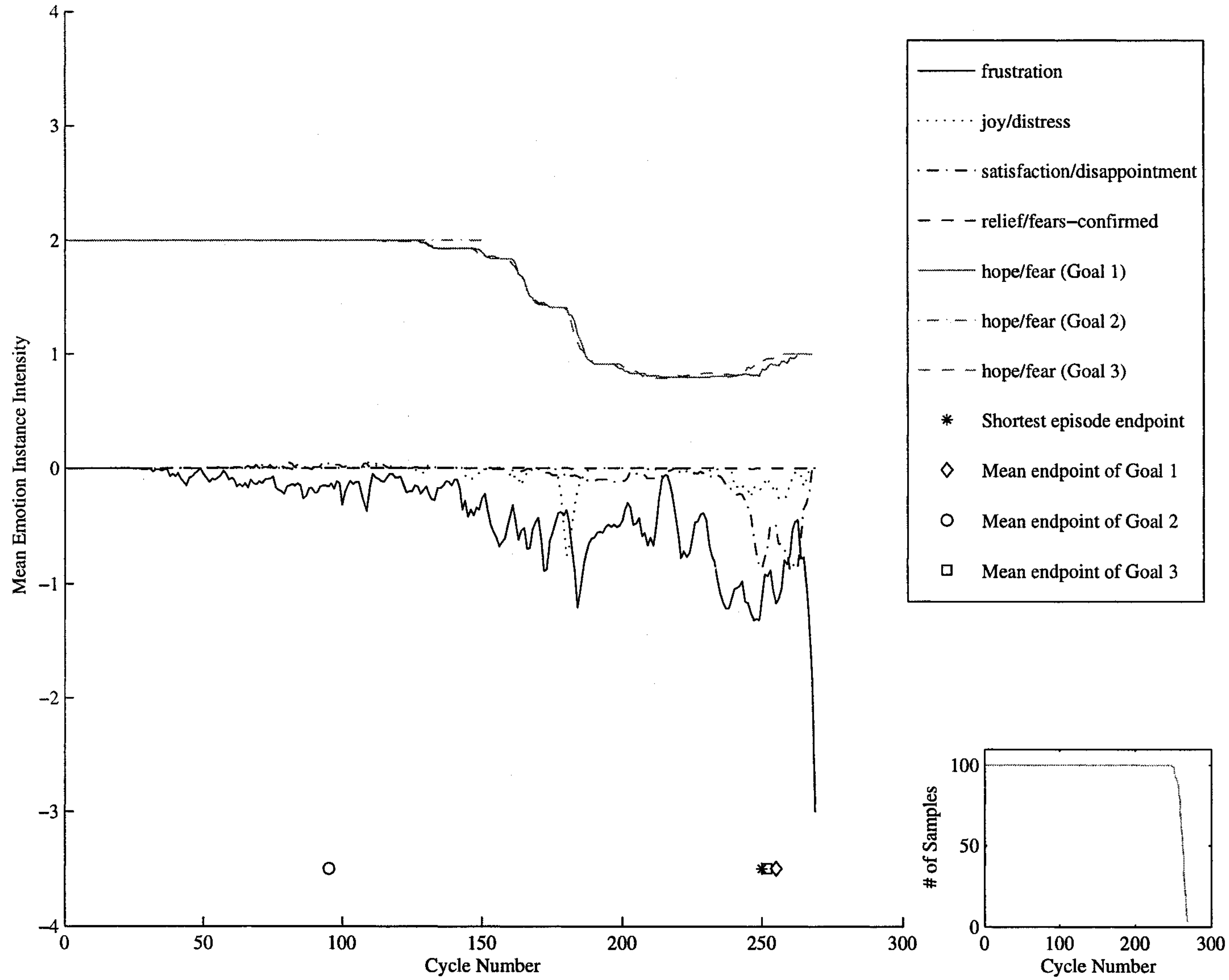

Figure A.32: Emotion chart for the noisy emotional agent in the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

A.1.3 Persistence

For each test combination, we present mean persistence and normalized mean persistence. All test combinations use concurrent goals.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.31	5.00	5.13
Rational (noisy)	1.25	1.25	1.26
Emotional (non-noisy)	4.10	4.24	4.07
Emotional (noisy)	1.76	1.69	1.72

Table A.17: Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	0.44	0.42	0.43
Rational (noisy)	0.10	0.10	0.11
Emotional (non-noisy)	0.34	0.35	0.34
Emotional (noisy)	0.15	0.14	0.14

Table A.18: Normalized mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.00	5.00	5.00
Rational (noisy)	1.20	1.18	1.19
Emotional (non-noisy)	2.73	2.75	2.72
Emotional (noisy)	1.20	1.23	1.22

Table A.19: Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1.00	1.00	1.00
Rational (noisy)	0.24	0.24	0.24
Emotional (non-noisy)	0.55	0.55	0.54
Emotional (noisy)	0.24	0.25	0.24

Table A.20: Normalized mean persistence for the symmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.00	7.36	5.00
Rational (noisy)	1.23	2.71	1.25
Emotional (non-noisy)	2.51	4.72	2.31
Emotional (noisy)	1.20	2.53	1.20

Table A.21: Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1.00	0.61	1.00
Rational (noisy)	0.25	0.23	0.25
Emotional (non-noisy)	0.50	0.39	0.46
Emotional (noisy)	0.24	0.21	0.24

Table A.22: Normalized mean persistence for the symmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.66	5.00	5.50
Rational (noisy)	1.46	1.28	1.47
Emotional (non-noisy)	4.03	2.27	3.91
Emotional (noisy)	2.17	1.21	2.11

Table A.23: Mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	0.47	1.00	0.46
Rational (noisy)	0.12	0.26	0.12
Emotional (non-noisy)	0.34	0.45	0.33
Emotional (noisy)	0.18	0.24	0.18

Table A.24: Normalized mean persistence for the symmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.08	5.06	5.29
Rational (noisy)	1.30	1.27	1.29
Emotional (non-noisy)	2.64	2.70	2.66
Emotional (noisy)	1.28	1.30	1.29

Table A.25: Mean persistence for the symmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	0.42	0.42	0.44
Rational (noisy)	0.11	0.11	0.11
Emotional (non-noisy)	0.22	0.22	0.22
Emotional (noisy)	0.11	0.11	0.11

Table A.26: Normalized mean persistence for the symmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.00	5.00	5.00
Rational (noisy)	1.22	1.22	1.23
Emotional (non-noisy)	3.01	2.89	3.11
Emotional (noisy)	1.19	1.26	1.21

Table A.27: Mean persistence for the symmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1.00	1.00	1.00
Rational (noisy)	0.24	0.24	0.25
Emotional (non-noisy)	0.60	0.58	0.62
Emotional (noisy)	0.24	0.25	0.24

Table A.28: Normalized mean persistence for the symmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.00	7.14	5.00
Rational (noisy)	1.18	2.62	1.17
Emotional (non-noisy)	2.98	5.06	3.14
Emotional (noisy)	1.19	2.61	1.25

Table A.29: Mean persistence for the symmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1.00	0.60	1.00
Rational (noisy)	0.24	0.22	0.23
Emotional (non-noisy)	0.60	0.42	0.63
Emotional (noisy)	0.24	0.22	0.25

Table A.30: Normalized mean persistence for the symmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.53	5.00	5.53
Rational (noisy)	1.53	1.29	1.49
Emotional (non-noisy)	2.94	3.11	3.01
Emotional (noisy)	1.51	1.27	1.53

Table A.31: Mean persistence for the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	0.46	1.00	0.46
Rational (noisy)	0.13	0.26	0.12
Emotional (non-noisy)	0.25	0.62	0.25
Emotional (noisy)	0.13	0.25	0.13

Table A.32: Normalized mean persistence for the symmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

A.2 Asymmetric Scenario

A.2.1 Plan Selection Counts

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-3-2 (100)	1-3-2 (100)	1-3-2 (100)
Rational (noisy)	3-1-2 (57)	3-1-2 (57)	3-1-2 (53)
	1-3-2 (42)	1-3-2 (39)	1-3-2 (47)
Emotional (non-noisy)	1-2-3 (100)	1-2-3 (50)	3-2-1 (21)
		1-3-2 (50)	2-3-1 (20)
Emotional (noisy)	3-1-2 (50)	1-2-3 (35)	1-3-2 (20)
	1-2-3 (36)	1-3-2 (28)	3-2-1 (18)

Table A.33: Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1 (100)	1 (100)
Rational (noisy)	1 (52)	3 (54)	1 (53)
	3 (48)	1 (46)	3 (46)
Emotional (non-noisy)	1 (100)	1 (45)	1 (43)
		2 (28)	2 (38)
Emotional (noisy)	1 (50)	3 (44)	1 (54)
	3 (50)	1 (40)	2 (28)

Table A.34: Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1-3-2 (100)	1 (100)
Rational (noisy)	3 (52)	1-3-2 (51)	1 (50)
	1 (48)	3-1-2 (46)	3 (50)
Emotional (non-noisy)	1 (100)	1-2-3 (30)	1 (67)
		2-1-3 (29)	2 (33)
Emotional (noisy)	3 (53)	1-2-3 (39)	1 (67)
	1 (47)	3-1-2 (31)	2 (30)

Table A.35: Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-3-2 (100)	1 (100)	1-3-2 (100)
Rational (noisy)	3-1-2 (54)	1 (52)	1-3-2 (52)
	1-3-2 (44)	3 (48)	3-1-2 (47)
Emotional (non-noisy)	1-2-3 (100)	1 (100)	3-1-2 (51)
			3-2-1 (49)
Emotional (noisy)	1-2-3 (45)	1 (57)	3-1-2 (49)
	3-1-2 (41)	2 (33)	3-2-1 (32)

Table A.36: Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-3-2 (100)	1-3-2 (100)	1-3-2 (100)
Rational (noisy)	3-1-2 (51)	1-3-2 (52)	1-3-2 (52)
	1-3-2 (47)	3-1-2 (48)	3-1-2 (47)
Emotional (non-noisy)	1-2-3 (90)	1-3-2 (93)	1-2-3 (76)
	1-3-2 (10)	1-2-3 (7)	1-3-2 (24)
Emotional (noisy)	3-1-2 (49)	1-2-3 (52)	1-2-3 (53)
	1-2-3 (35)	1-3-2 (45)	1-3-2 (41)

Table A.37: Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1 (100)	1 (100)
Rational (noisy)	3 (55)	1 (54)	3 (53)
	1 (45)	3 (46)	1 (47)
Emotional (non-noisy)	1 (100)	1 (100)	1 (100)
Emotional (noisy)	3 (57)	1 (99)	1 (95)
	1 (43)	3 (1)	3 (4)

Table A.38: Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1-3-2 (100)	1 (100)
Rational (noisy)	1 (56)	1-3-2 (51)	3 (57)
	3 (44)	3-1-2 (46)	1 (43)
Emotional (non-noisy)	1 (100)	1-2-3 (97)	1 (100)
		1-3-2 (3)	
Emotional (noisy)	3 (56)	1-2-3 (73)	1 (95)
	1 (44)	1-3-2 (22)	3 (5)

Table A.39: Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-3-2 (100)	1 (100)	1-3-2 (100)
Rational (noisy)	3-1-2 (55)	1 (50)	1-3-2 (50)
	1-3-2 (43)	3 (49)	3-1-2 (48)
Emotional (non-noisy)	1-2-3 (96)	1 (100)	1-3-2 (98)
	1-3-2 (4)		1-2-3 (2)
Emotional (noisy)	3-1-2 (57)	1 (97)	1-3-2 (61)
	1-2-3 (33)	3 (3)	1-2-3 (38)

Table A.40: Plan selection counts (out of 100 episodes) for the asymmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-3-2 (100)	1-3-2 (100)	1-3-2 (100)
Rational (noisy)	3-1-2 (50)	1-3-2 (54)	3-1-2 (52)
	1-3-2 (48)	3-1-2 (45)	1-3-2 (47)
Emotional (non-noisy)	1-2-3 (62)	1-2-3 (67)	1-2-3 (63)
	1-3-2 (38)	1-3-2 (33)	1-3-2 (37)
Emotional (noisy)	1-2-3 (55)	1-2-3 (48)	1-2-3 (49)
	1-3-2 (29)	1-3-2 (36)	1-3-2 (33)

Table A.41: Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1 (100)	1 (100)
Rational (noisy)	3 (51)	3 (54)	3 (55)
	1 (49)	1 (46)	1 (45)
Emotional (non-noisy)	1 (100)	1 (100)	1 (100)
Emotional (noisy)	1 (81)	1 (83)	1 (88)
	3 (19)	3 (17)	3 (12)

Table A.42: Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1-3-2 (100)	1 (100)
Rational (noisy)	1 (54)	3-1-2 (51)	1 (52)
	3 (46)	1-3-2 (47)	3 (48)
Emotional (non-noisy)	1 (100)	1-2-3 (100)	1 (100)
Emotional (noisy)	1 (74)	1-2-3 (77)	1 (82)
	3 (26)	1-3-2 (11)	3 (18)

Table A.43: Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-3-2 (100)	1 (100)	1-3-2 (100)
Rational (noisy)	3-1-2 (55)	1 (64)	3-1-2 (54)
	1-3-2 (45)	3 (36)	1-3-2 (44)
Emotional (non-noisy)	1-2-3 (79)	1 (100)	1-2-3 (76)
	1-3-2 (21)		1-3-2 (24)
Emotional (noisy)	1-2-3 (57)	1 (88)	1-2-3 (70)
	1-3-2 (22)	3 (12)	3-1-2 (12)

Table A.44: Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-3-2 (100)	1-3-2 (100)	1-3-2 (100)
Rational (noisy)	1-3-2 (56)	3-1-2 (51)	3-1-2 (49)
	3-1-2 (43)	1-3-2 (45)	1-3-2 (48)
Emotional (non-noisy)	1-2-3 (57)	1-2-3 (58)	1-2-3 (68)
	1-3-2 (43)	1-3-2 (42)	1-3-2 (32)
Emotional (noisy)	1-2-3 (44)	1-2-3 (51)	1-2-3 (57)
	1-3-2 (36)	1-3-2 (30)	1-3-2 (31)

Table A.45: Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1 (100)	1 (100)
Rational (noisy)	3 (52)	1 (54)	1 (57)
	1 (48)	3 (46)	3 (43)
Emotional (non-noisy)	1 (100)	1 (100)	1 (100)
Emotional (noisy)	1 (79)	1 (78)	1 (85)
	3 (21)	3 (22)	3 (15)

Table A.46: Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1 (100)	1-3-2 (100)	1 (100)
Rational (noisy)	3 (60)	1-3-2 (58)	3 (54)
	1 (40)	3-1-2 (41)	1 (46)
Emotional (non-noisy)	1 (100)	1-2-3 (80)	1 (100)
		1-3-2 (20)	
Emotional (noisy)	1 (84)	1-2-3 (57)	1 (89)
	3 (16)	1-3-2 (26)	3 (11)

Table A.47: Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1-3-2 (100)	1 (100)	1-3-2 (100)
Rational (noisy)	3-1-2 (54)	1 (53)	3-1-2 (50)
	1-3-2 (43)	3 (47)	1-3-2 (47)
Emotional (non-noisy)	1-3-2 (65)	1 (100)	1-2-3 (54)
	1-2-3 (35)		1-3-2 (46)
Emotional (noisy)	1-2-3 (45)	1 (77)	1-2-3 (49)
	1-3-2 (39)	3 (23)	1-3-2 (31)

Table A.48: Plan selection counts (out of 100 episodes) for the asymmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

A.2.2 Emotion Charts

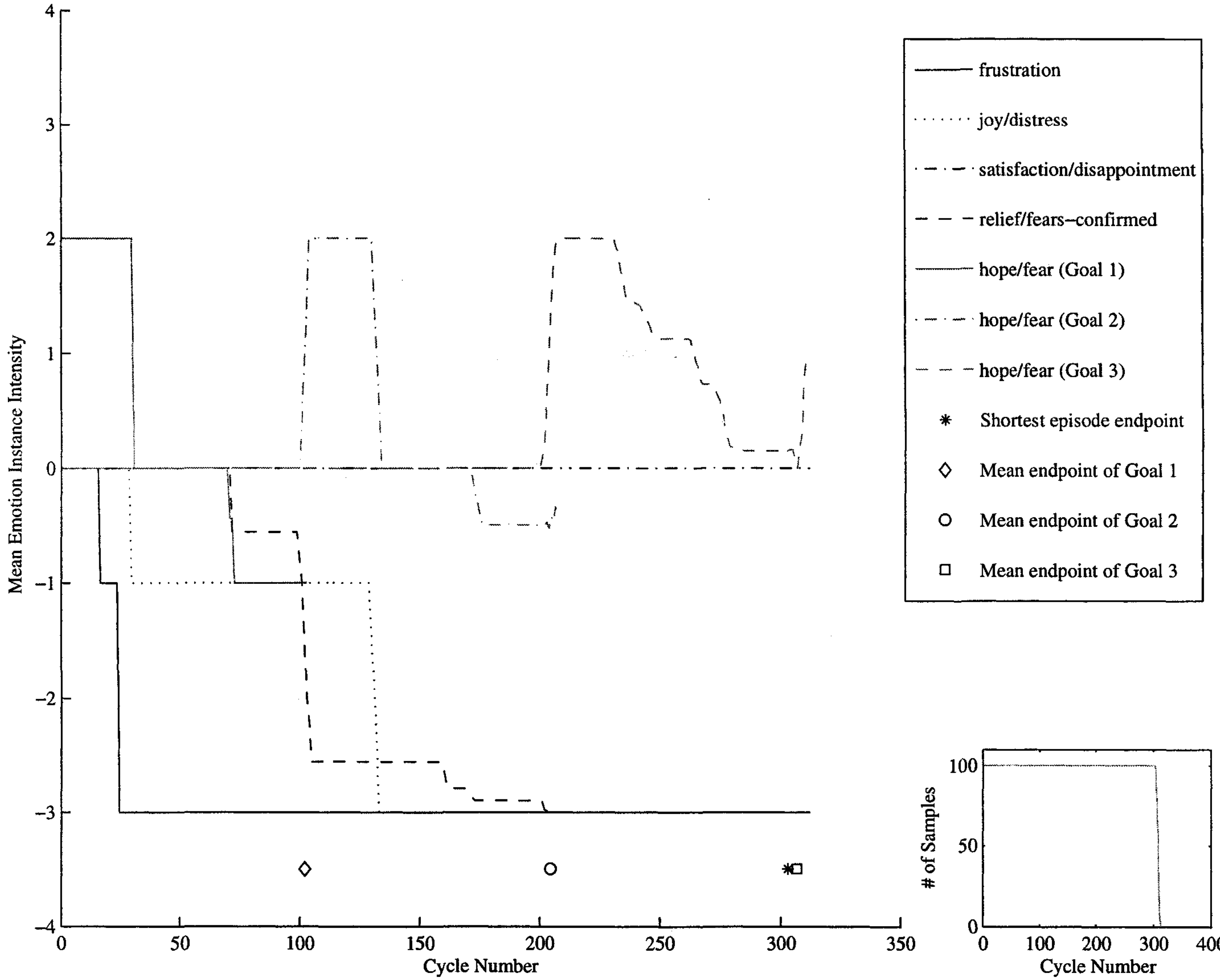

Figure A.33: Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

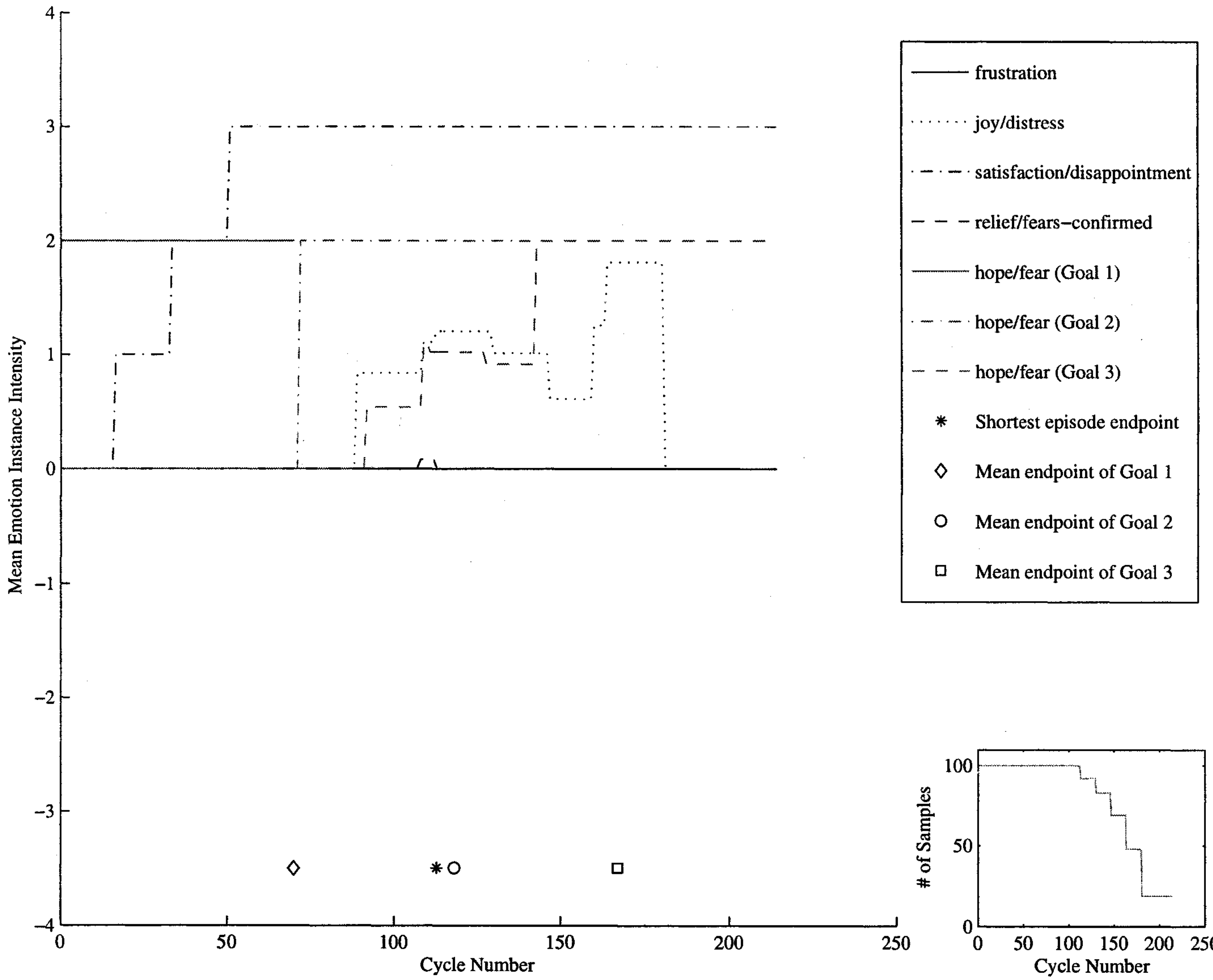

Figure A.34: Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience.

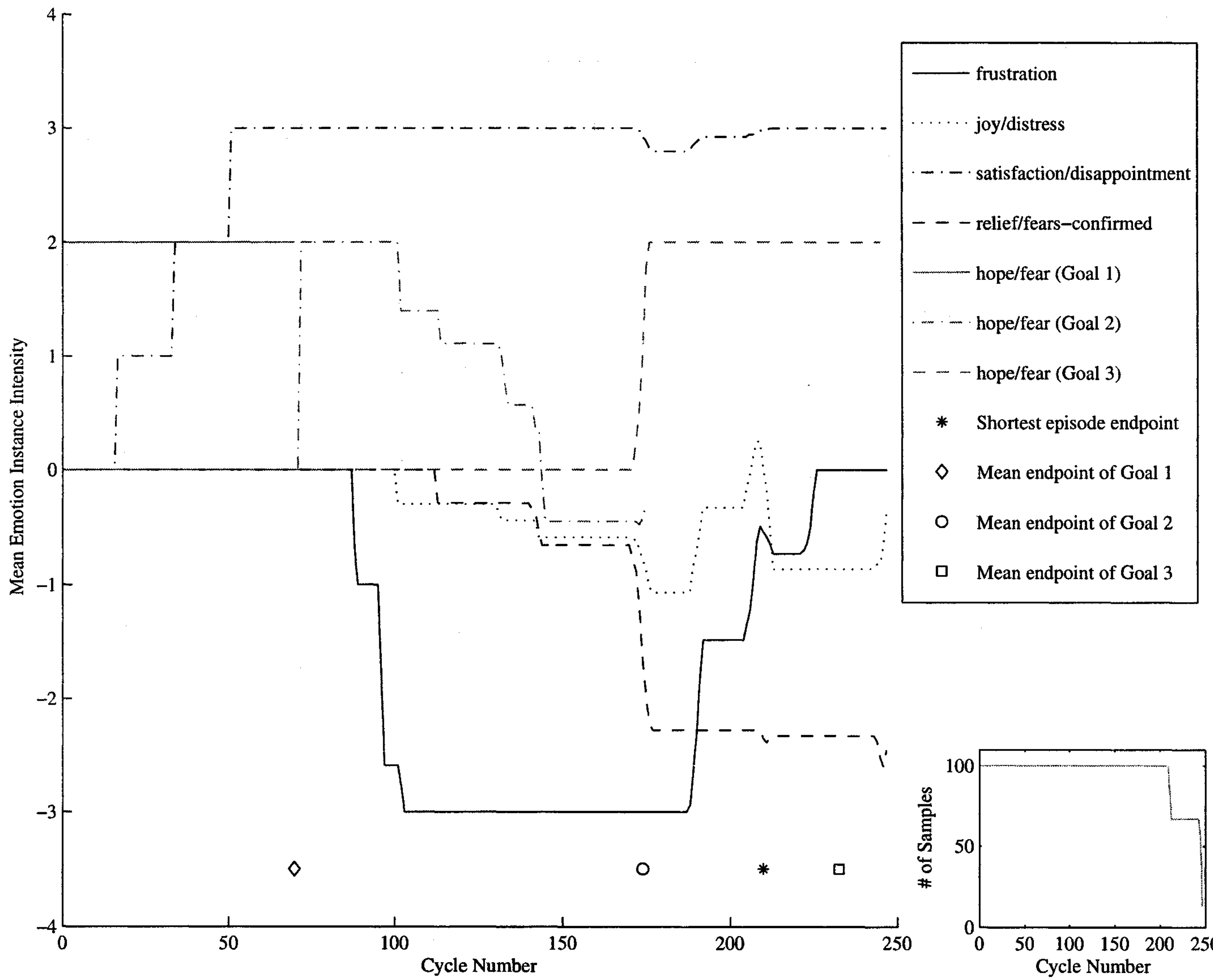

Figure A.35: Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

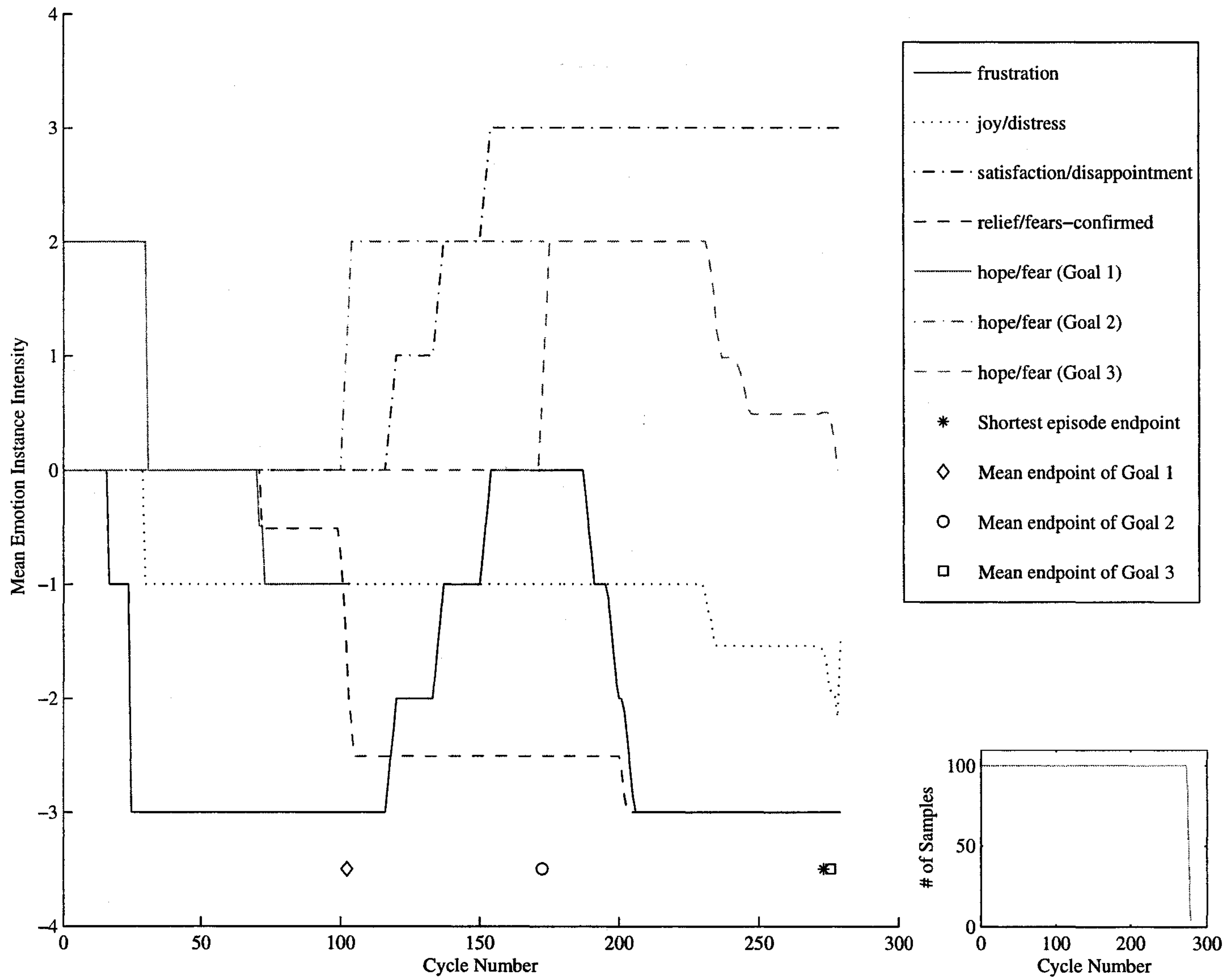

Figure A.36: Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

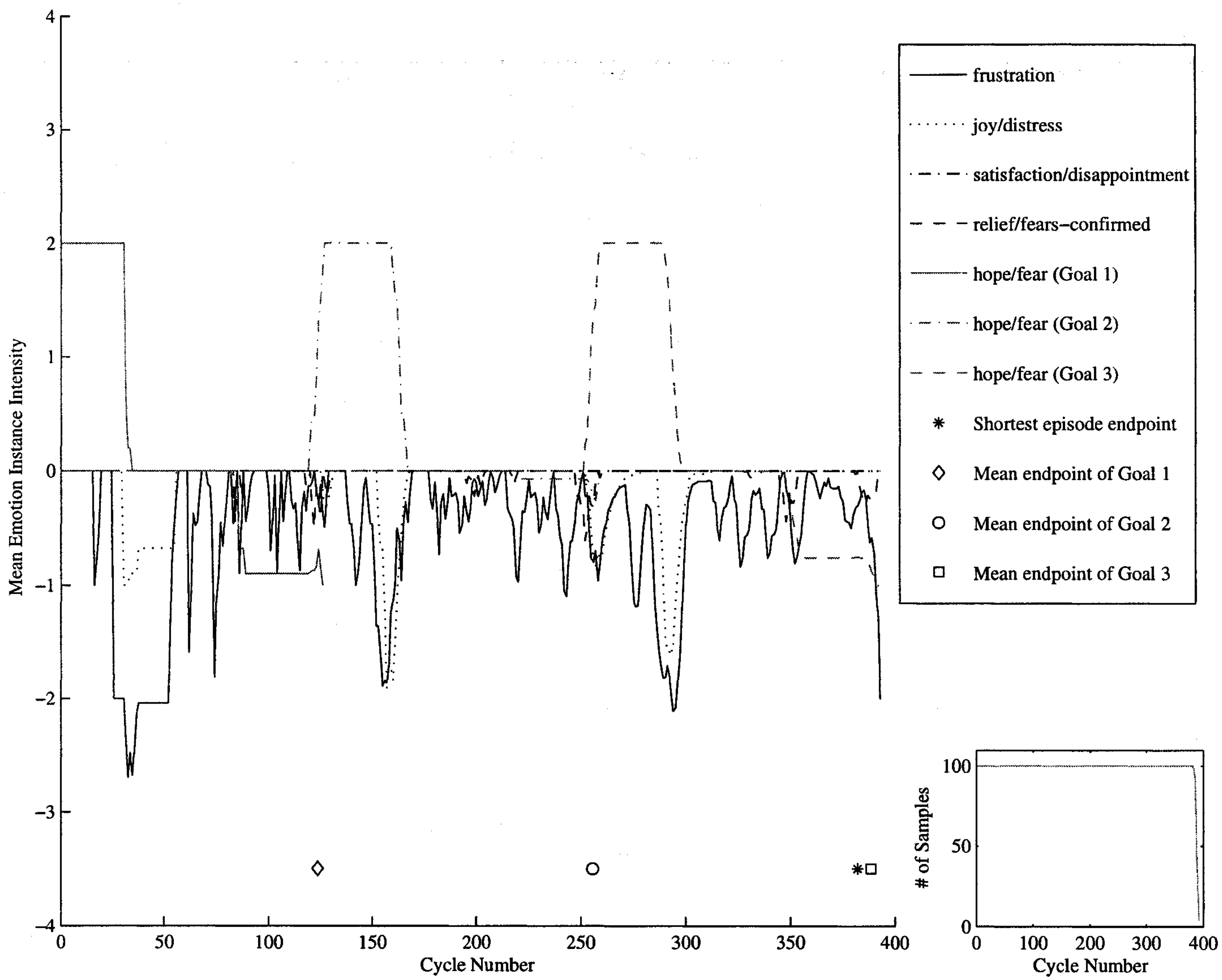

Figure A.37: Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

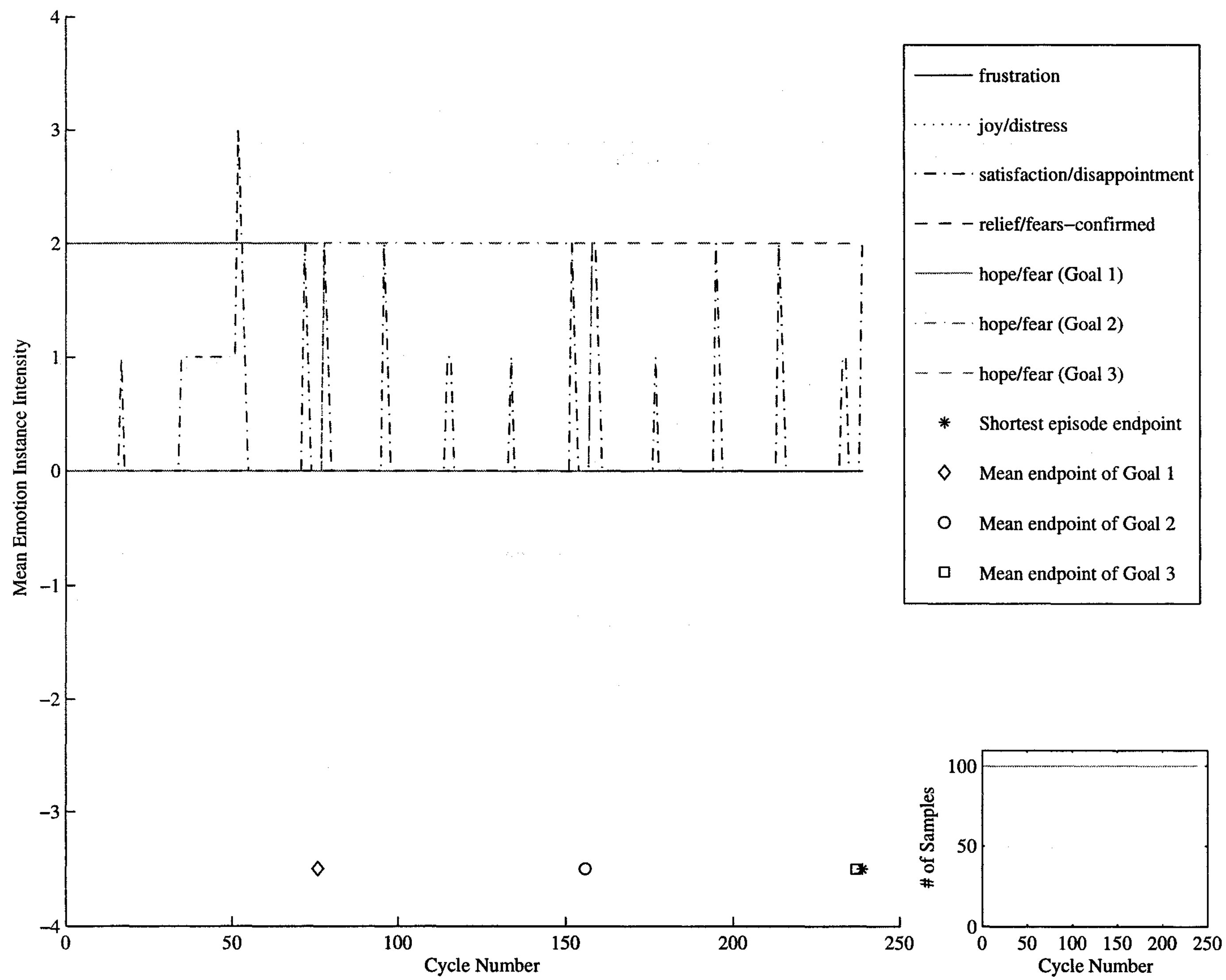

Figure A.38: Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience.

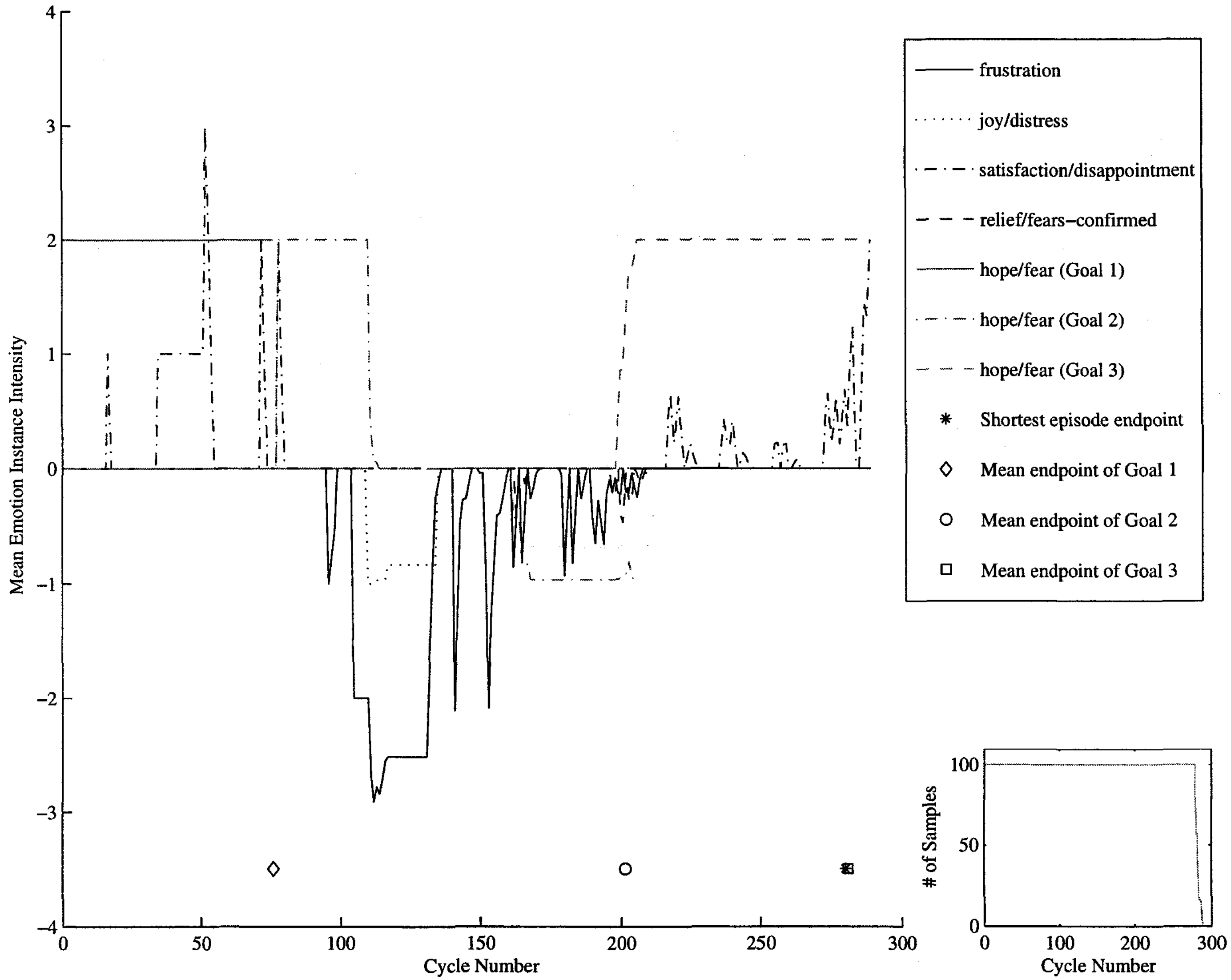

Figure A.39: Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience.

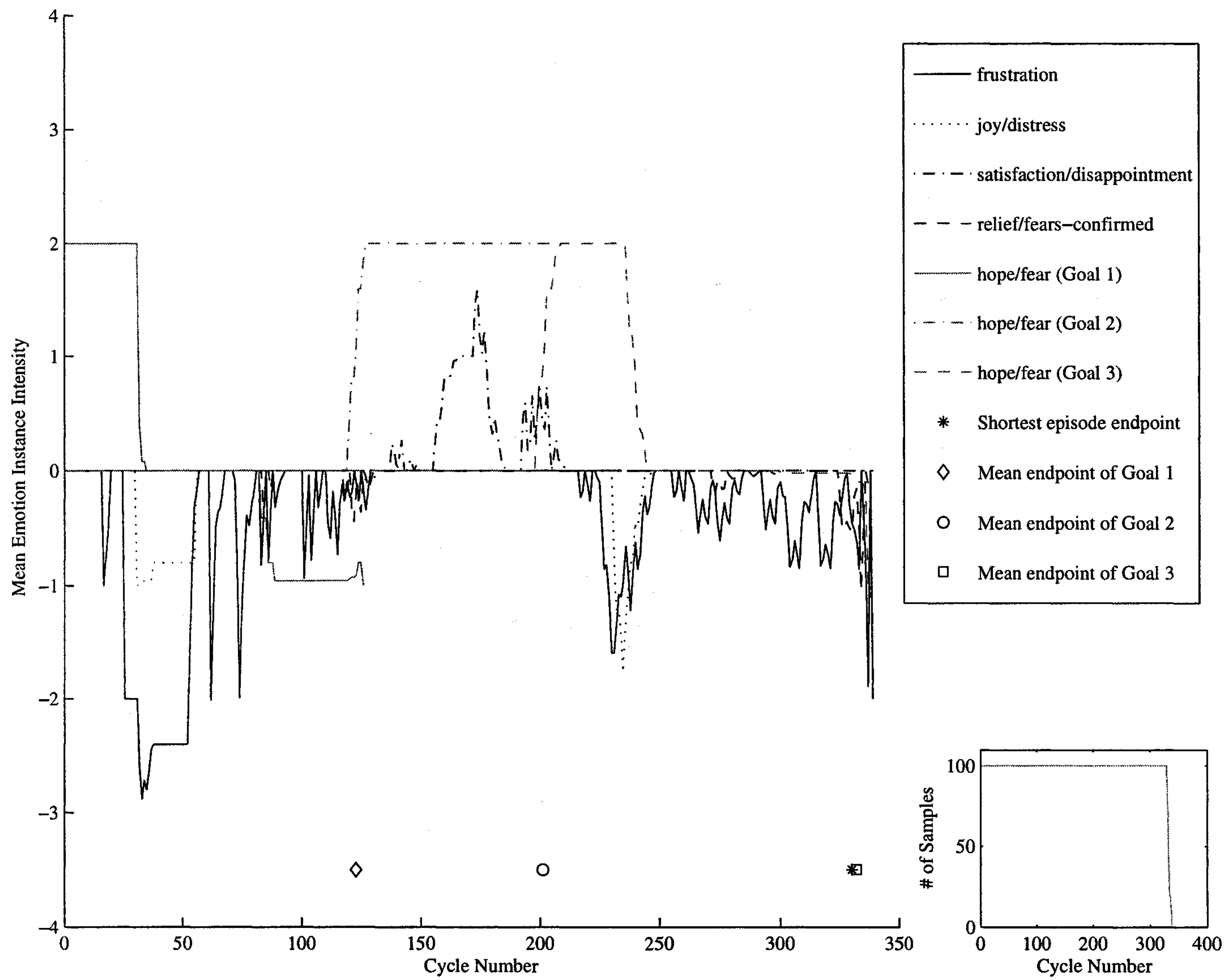

Figure A.40: Emotion chart for the non-noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

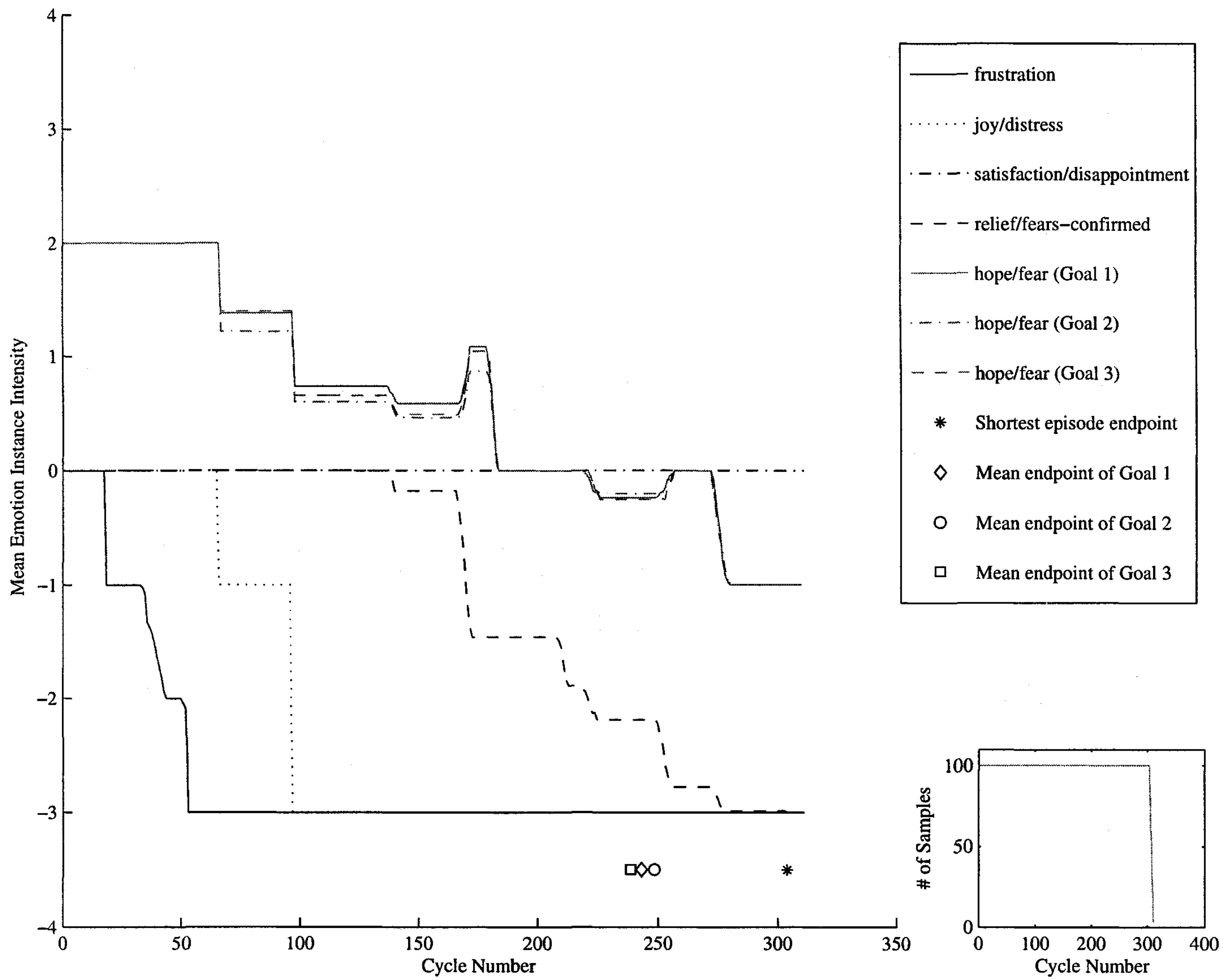

Figure A.41: Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

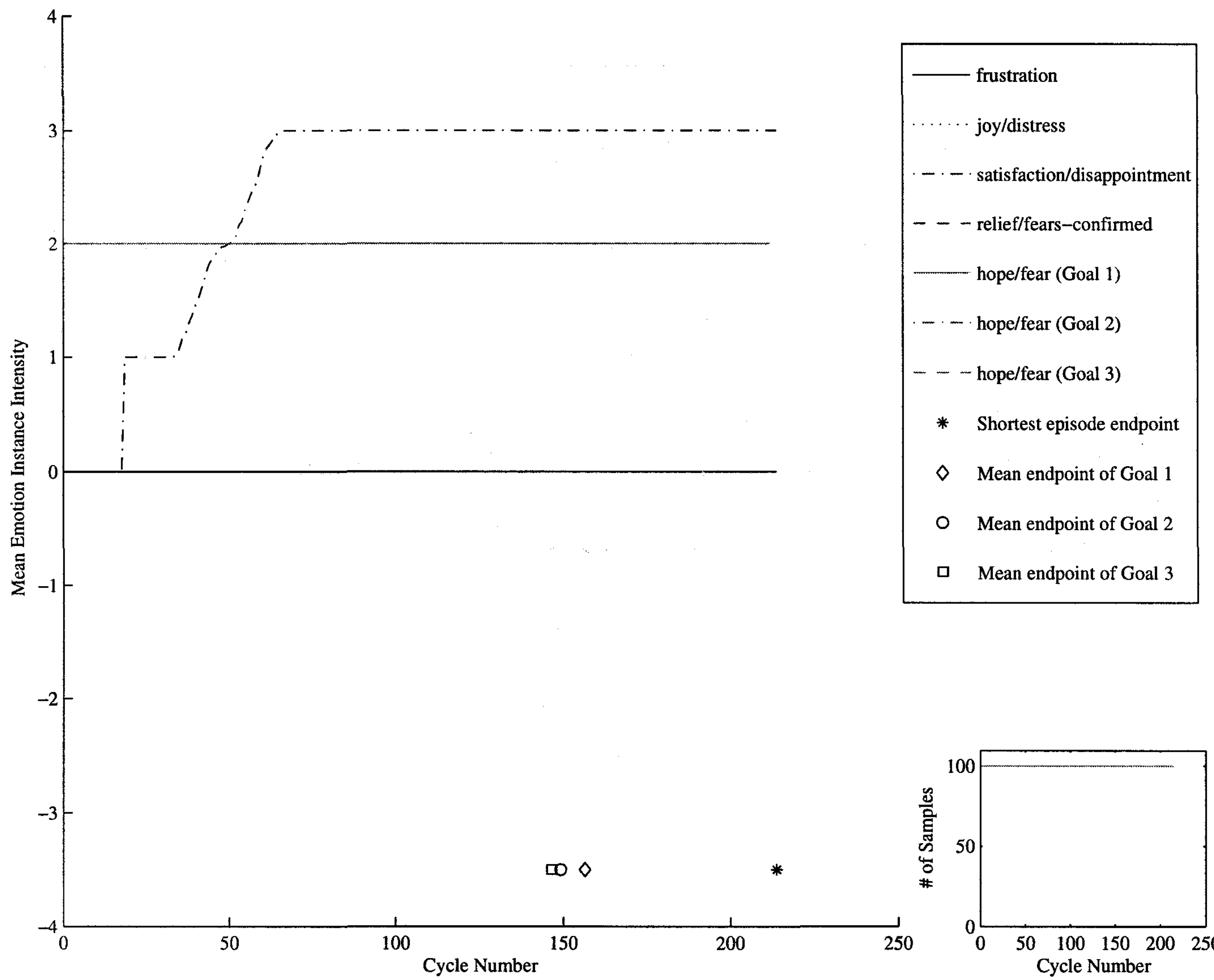

Figure A.42: Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience.

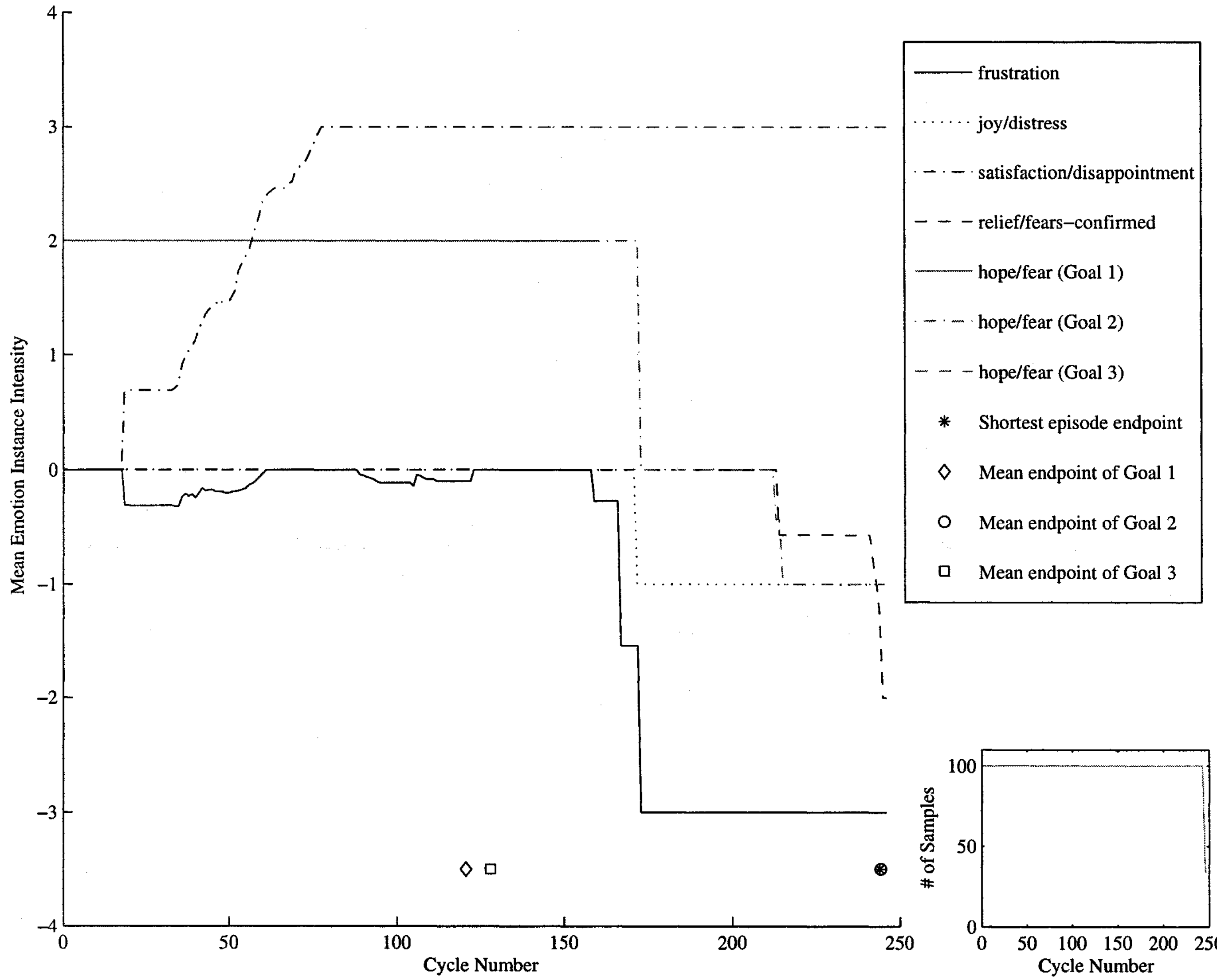

Figure A.43: Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

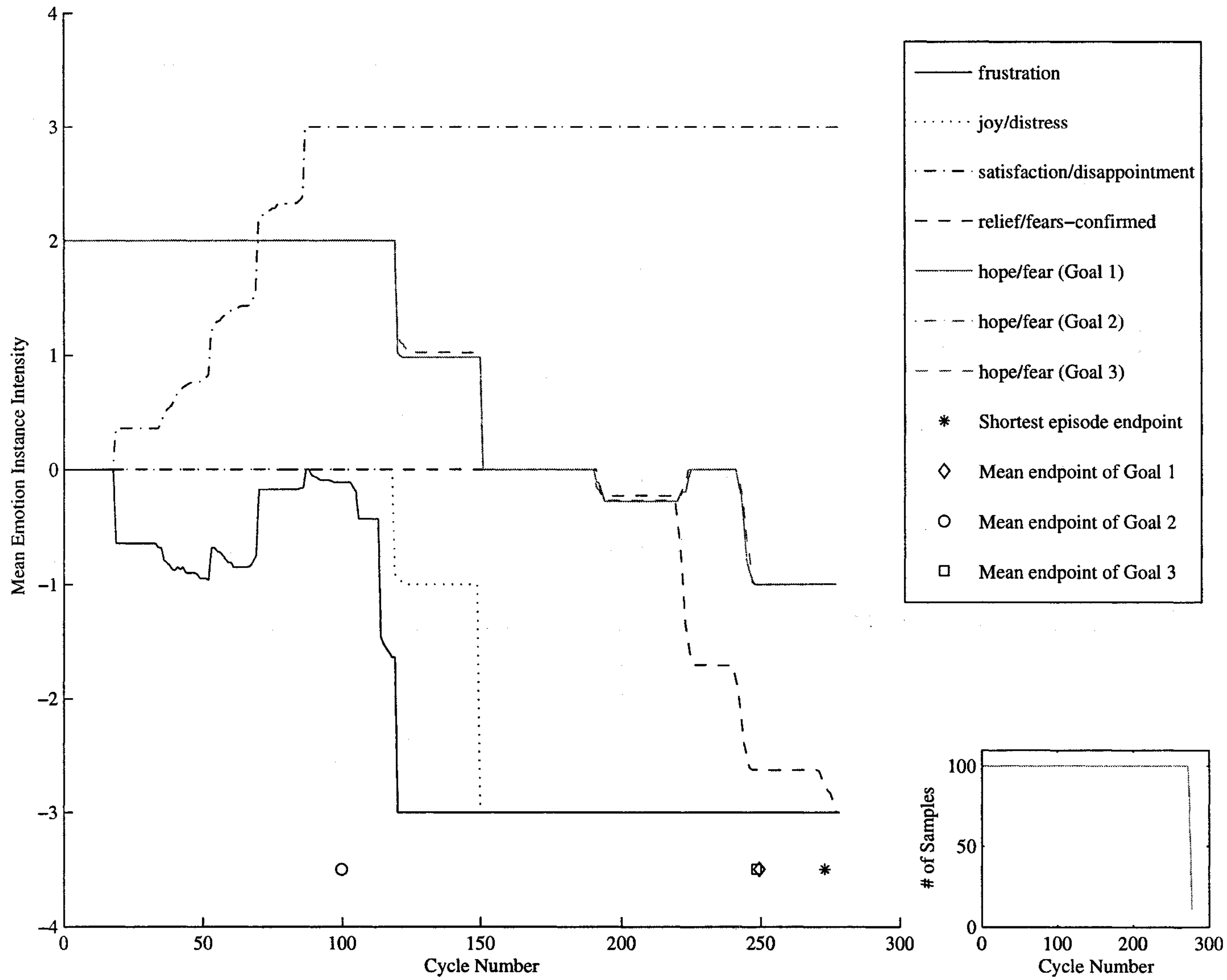

Figure A.44: Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

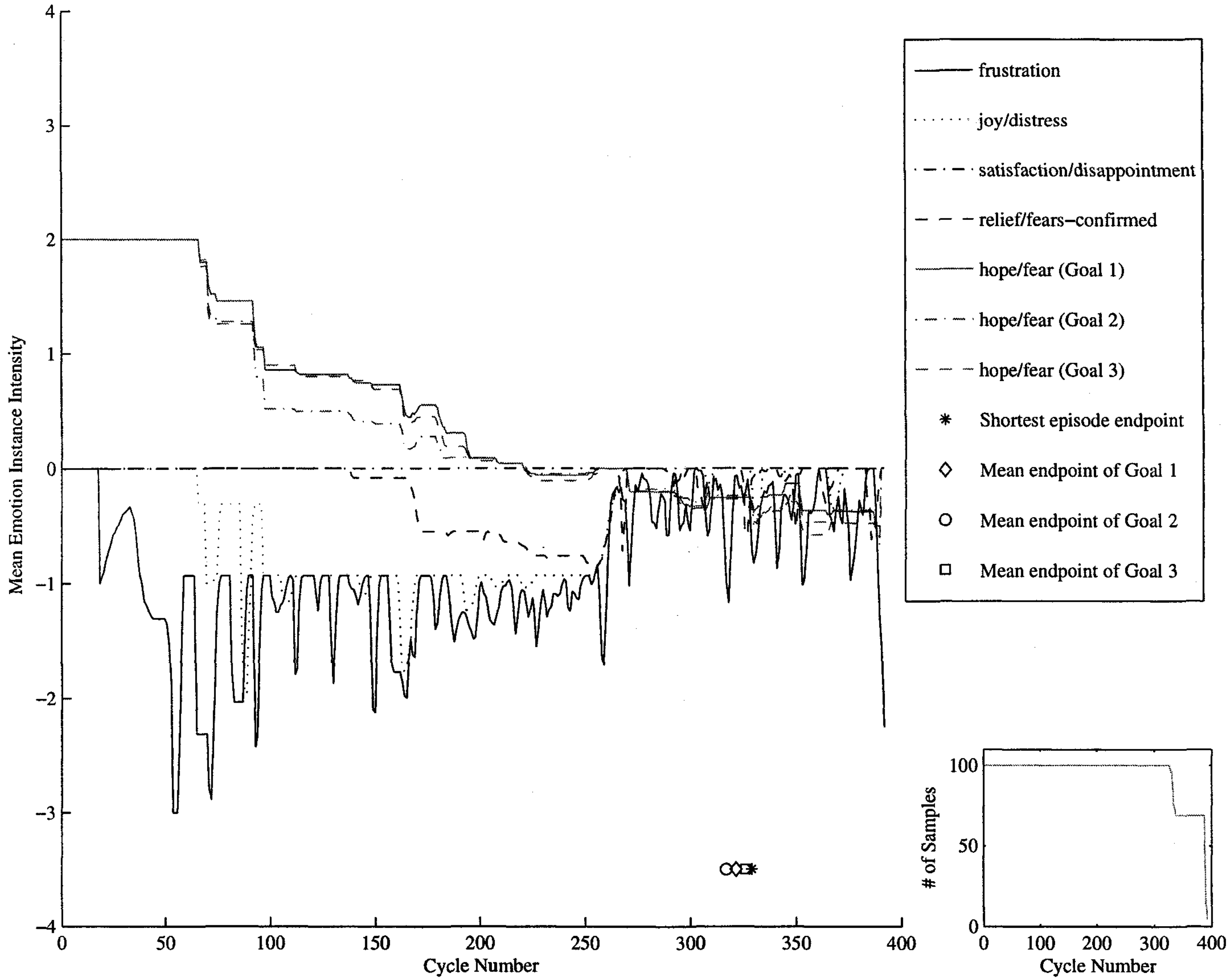

Figure A.45: Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

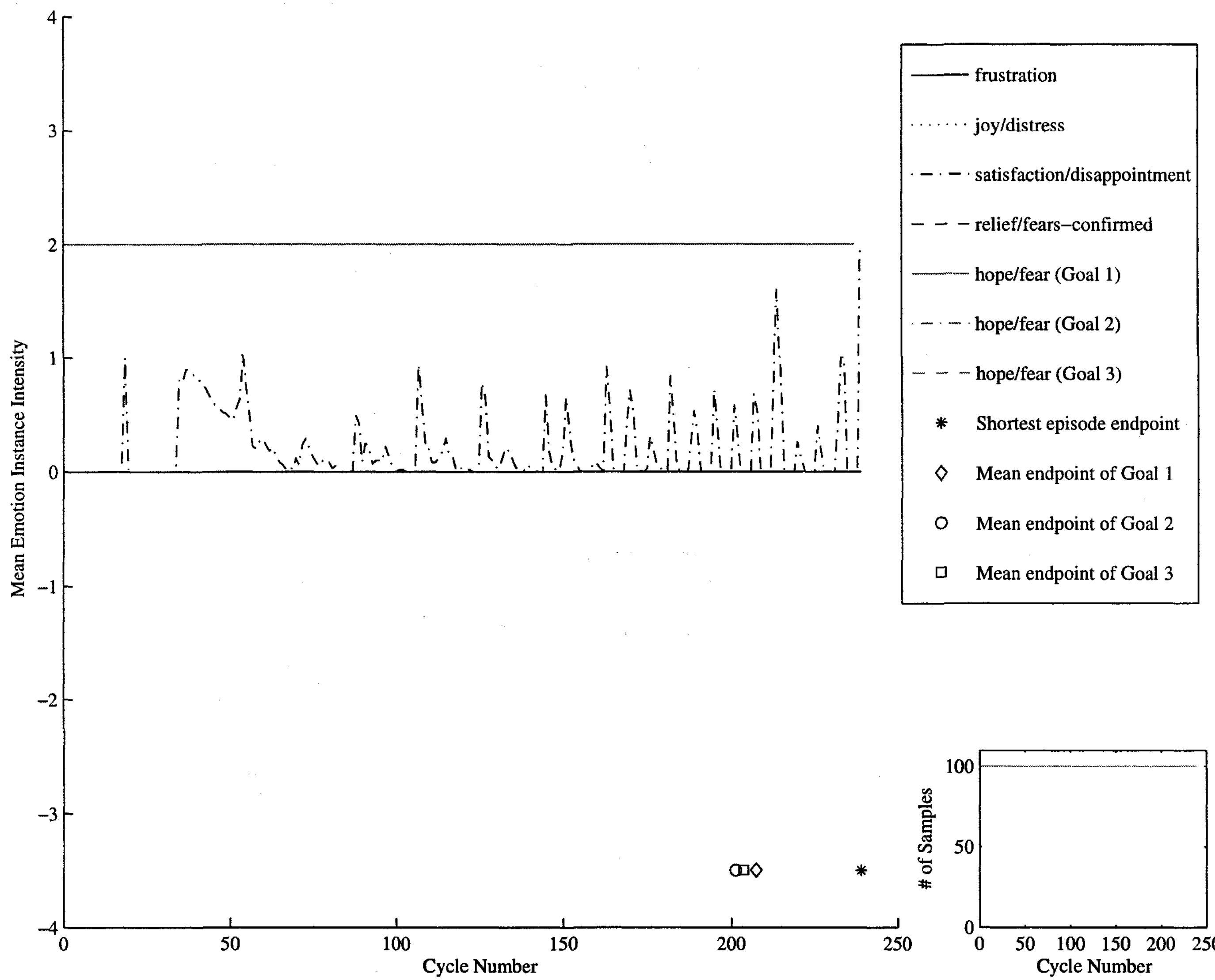

Figure A.46: Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience.

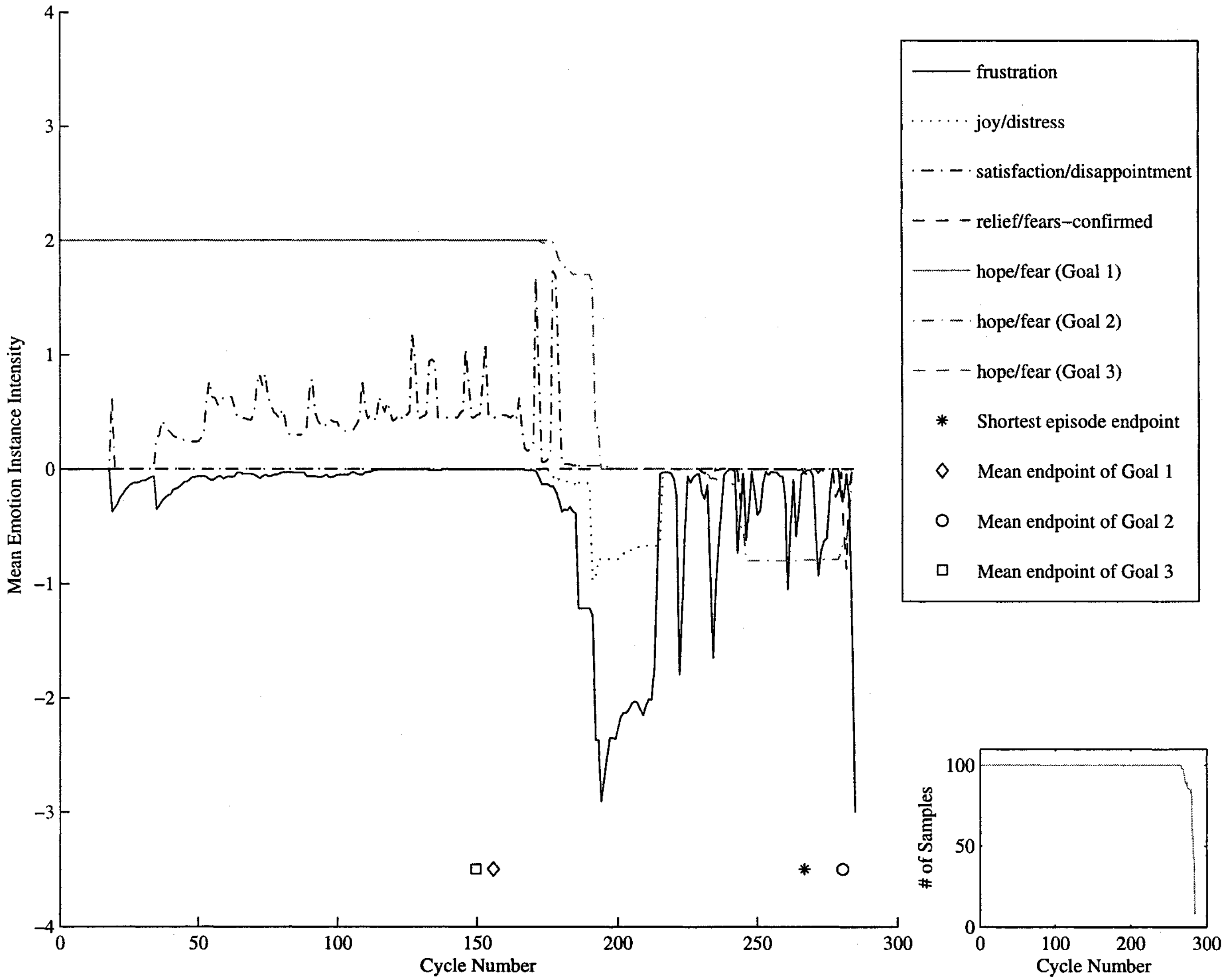

Figure A.47: Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience.

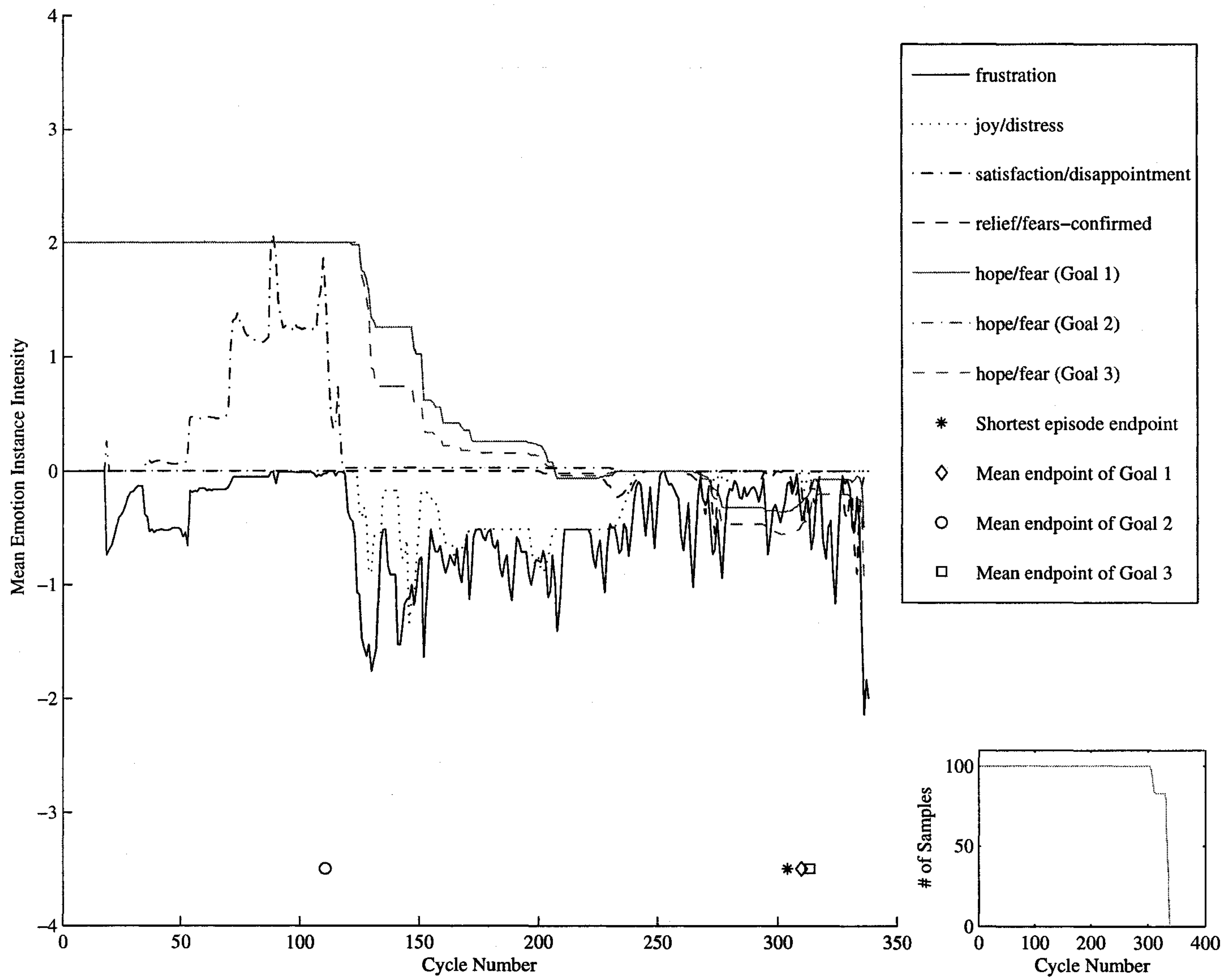

Figure A.48: Emotion chart for the non-noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

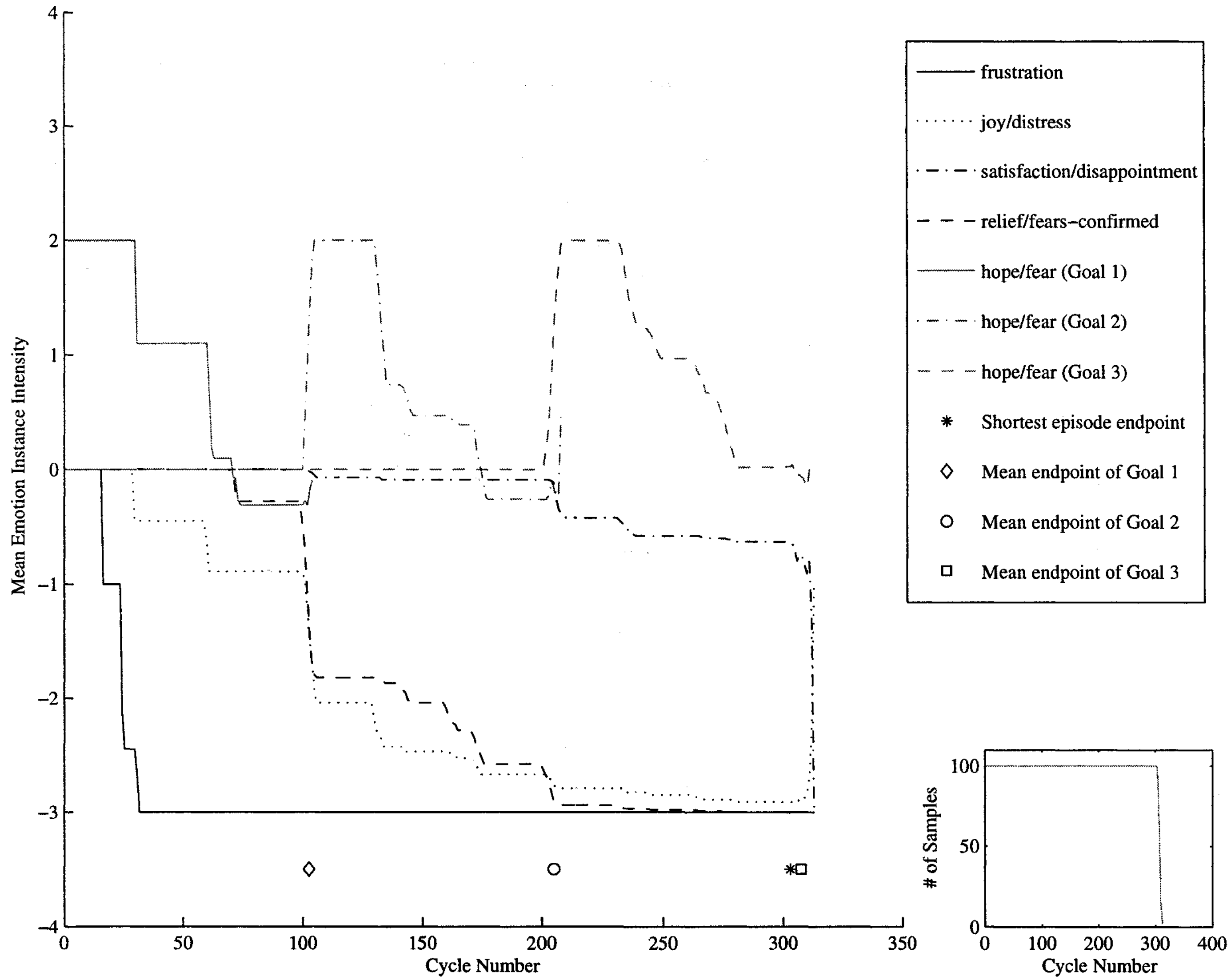

Figure A.49: Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

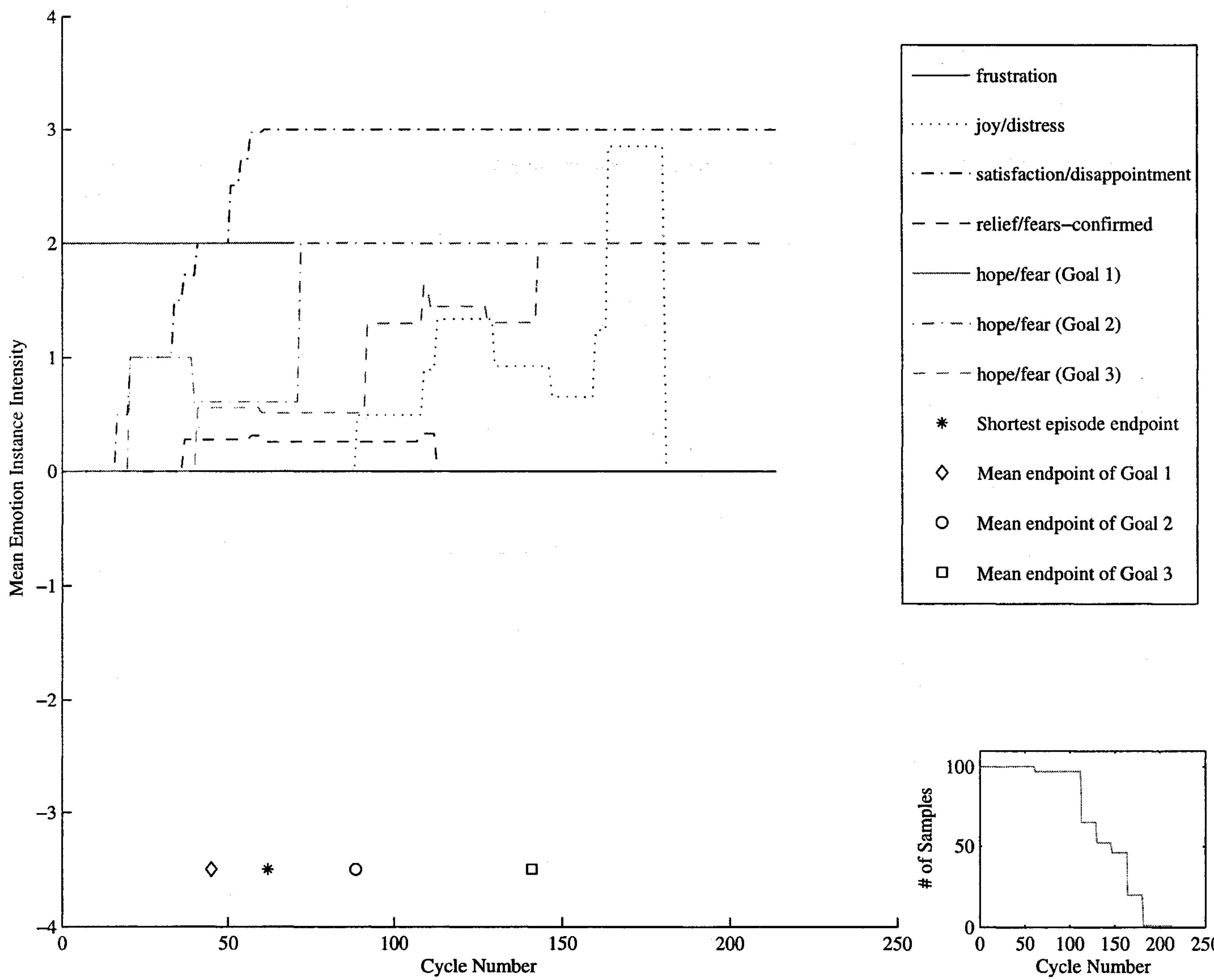

Figure A.50: Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Success-Success-Success problem solving experience.

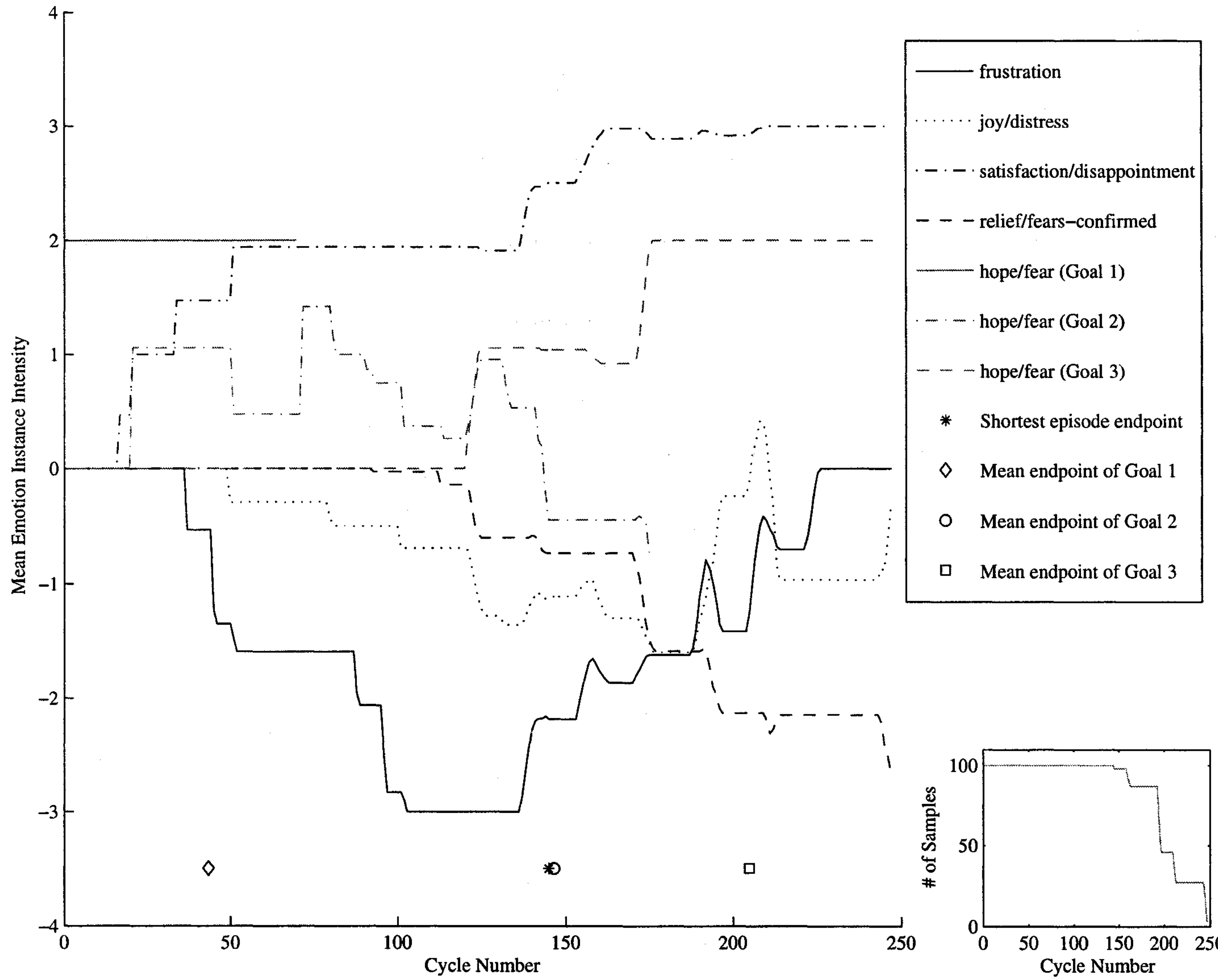

Figure A.51: Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

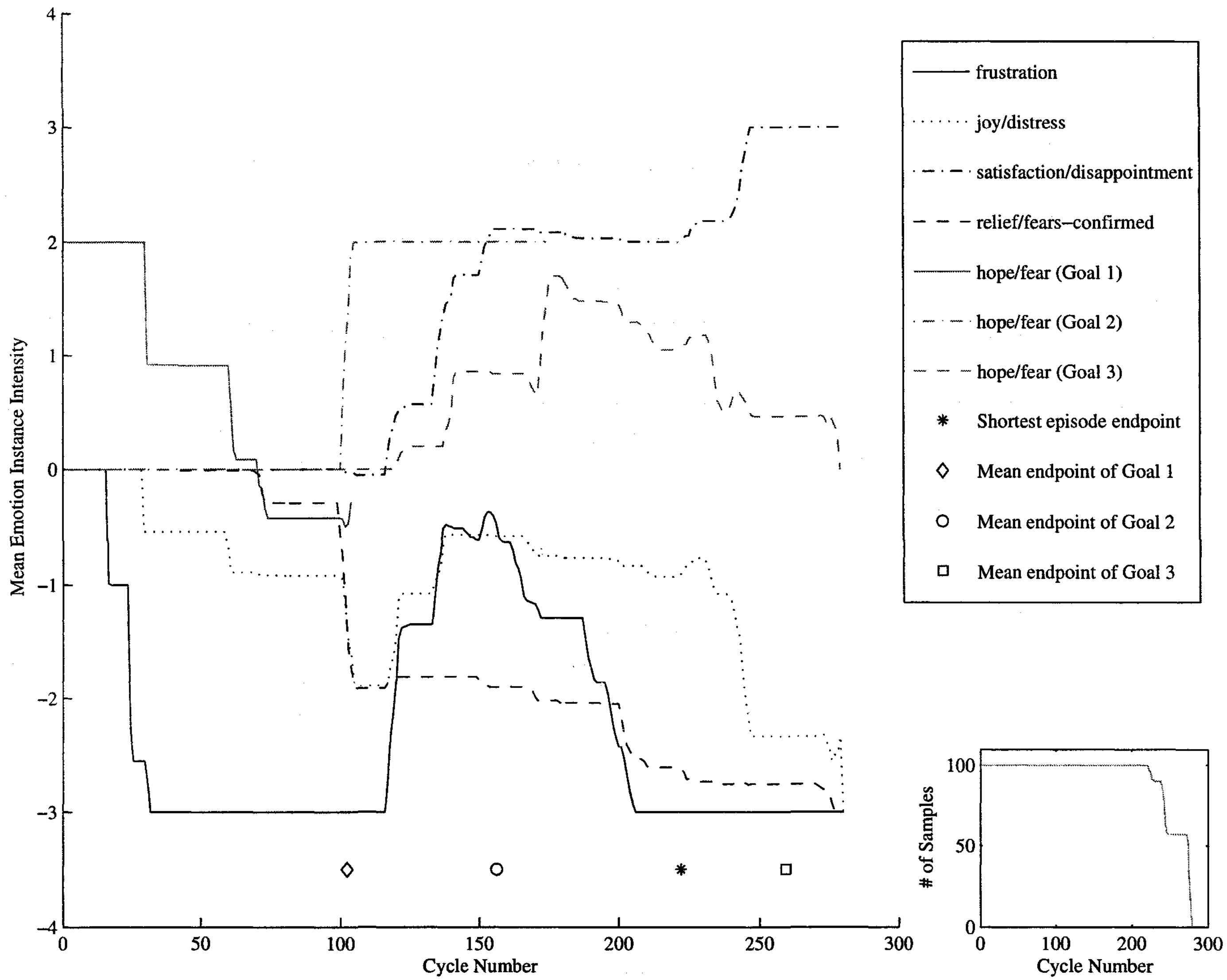

Figure A.52: Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

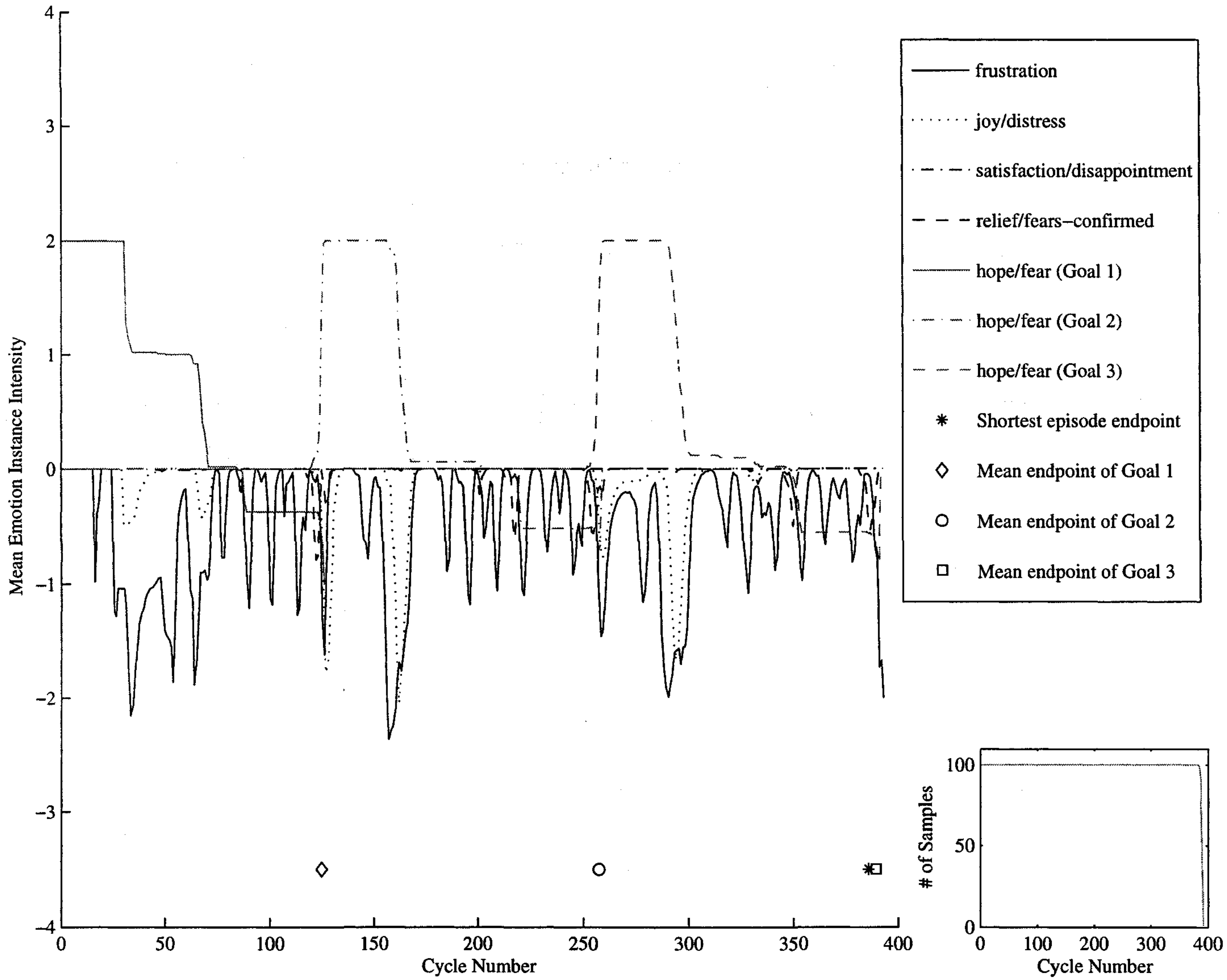

Figure A.53: Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

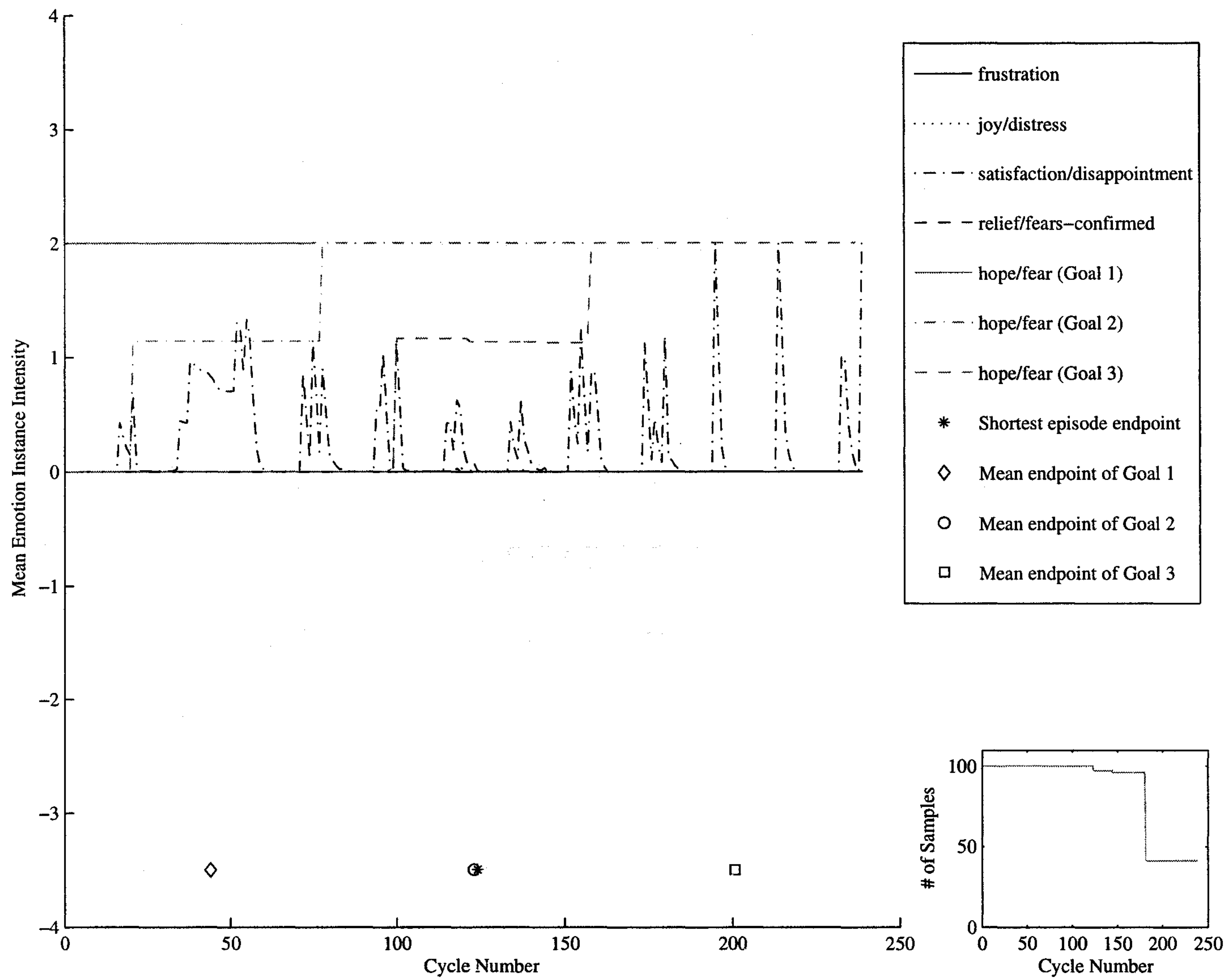

Figure A.54: Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Success-Success-Success problem solving experience.

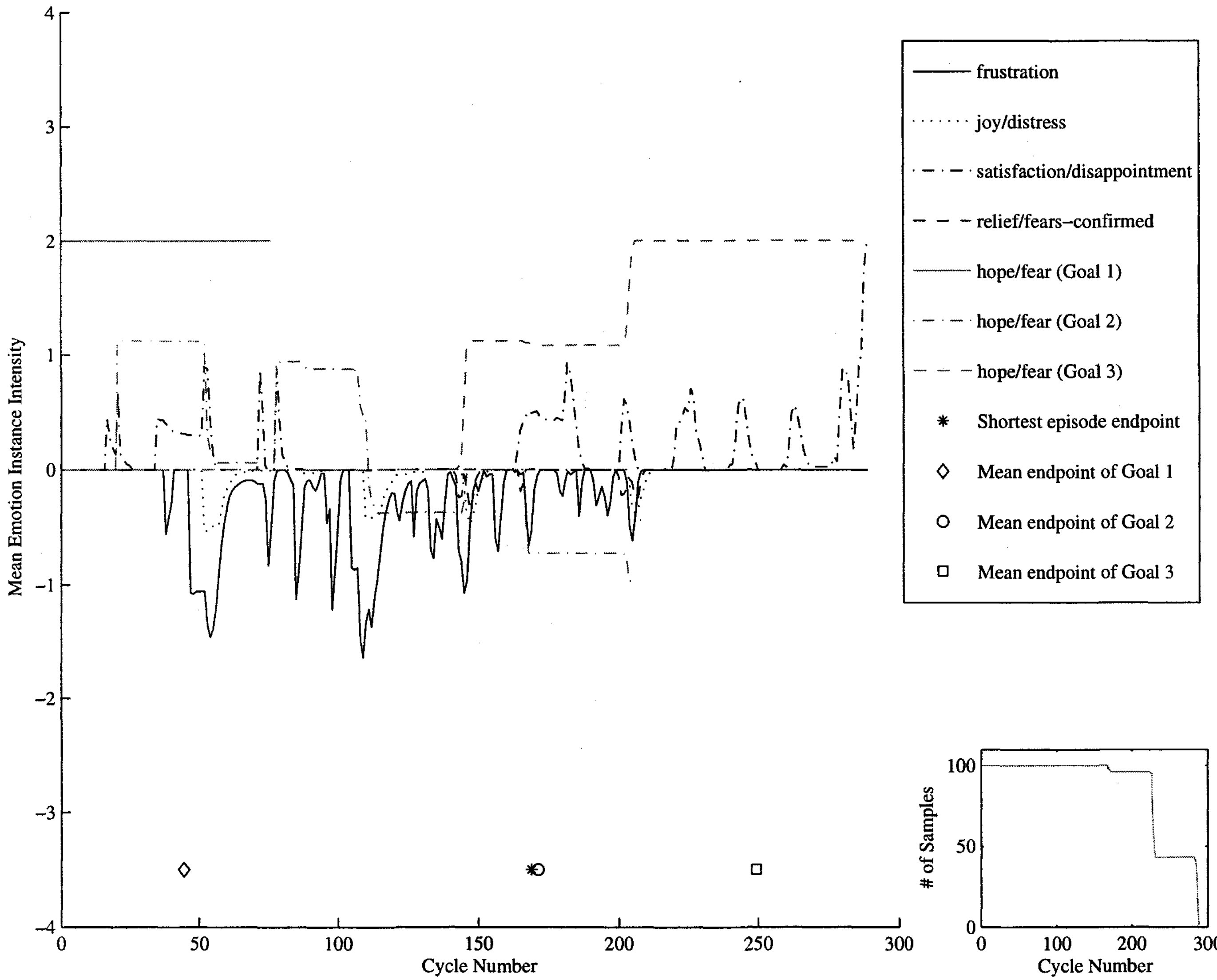

Figure A.55: Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Success-Failure-Success problem solving experience.

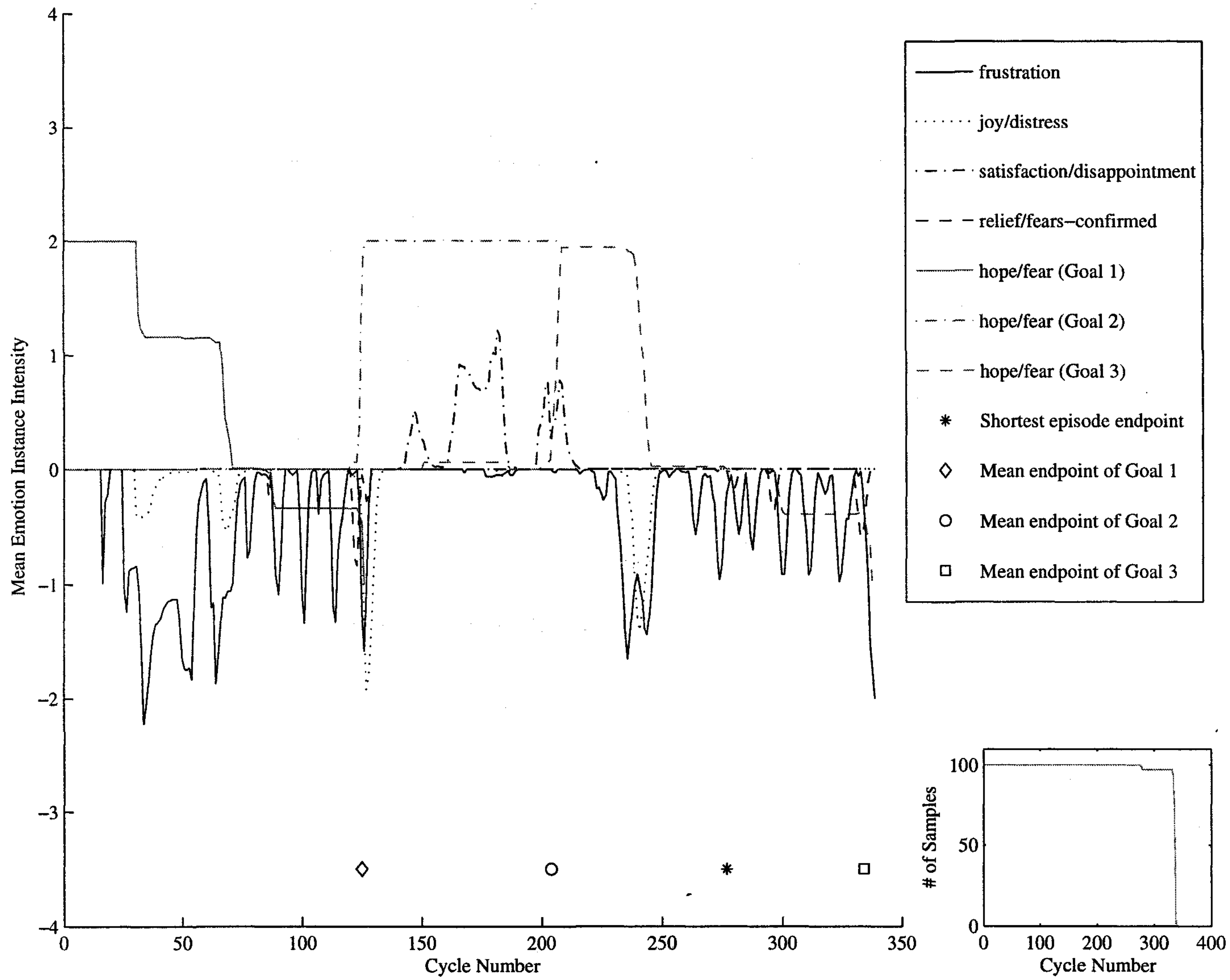

Figure A.56: Emotion chart for the noisy emotional agent in the asymmetric sequential scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

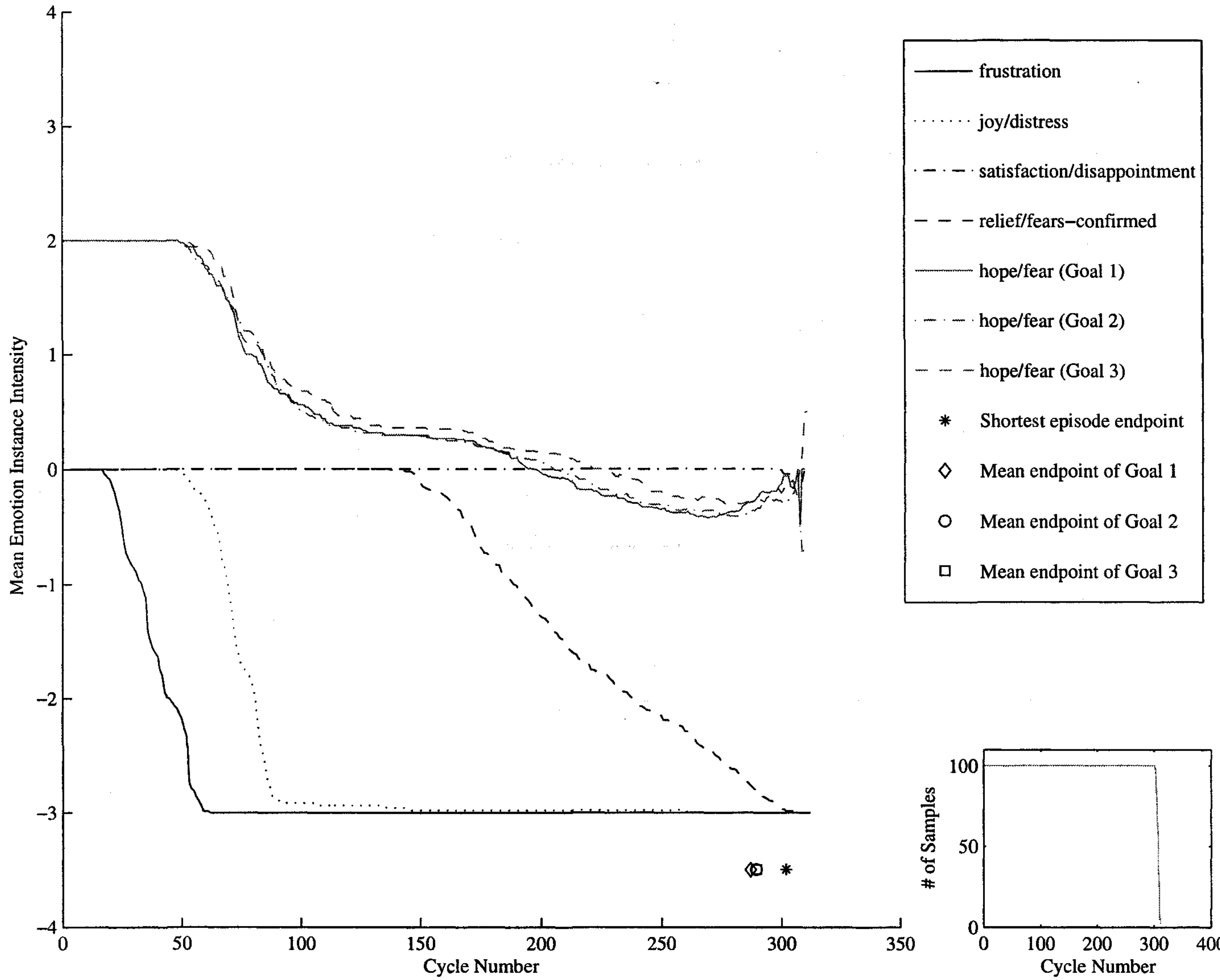

Figure A.57: Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

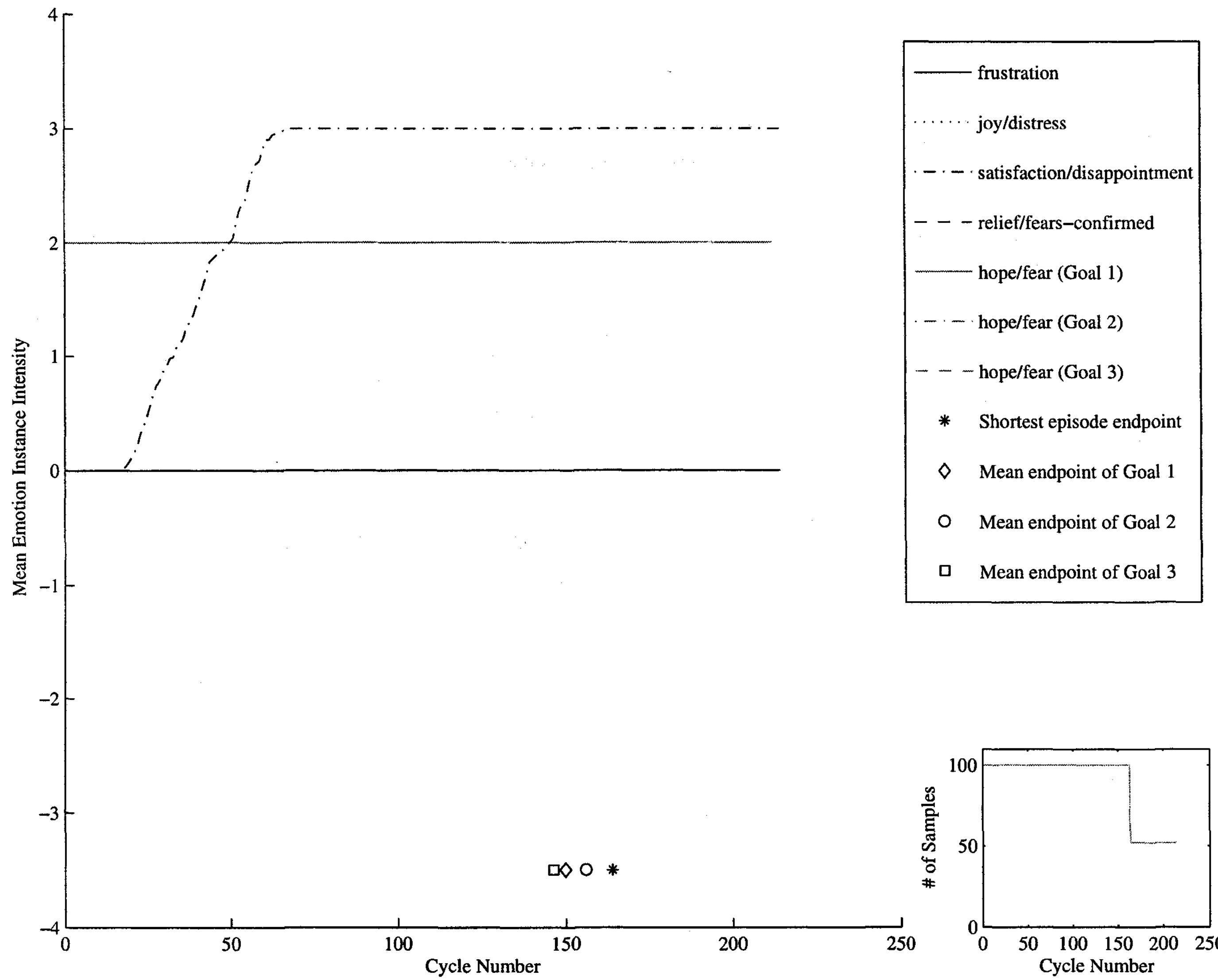

Figure A.58: Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience.

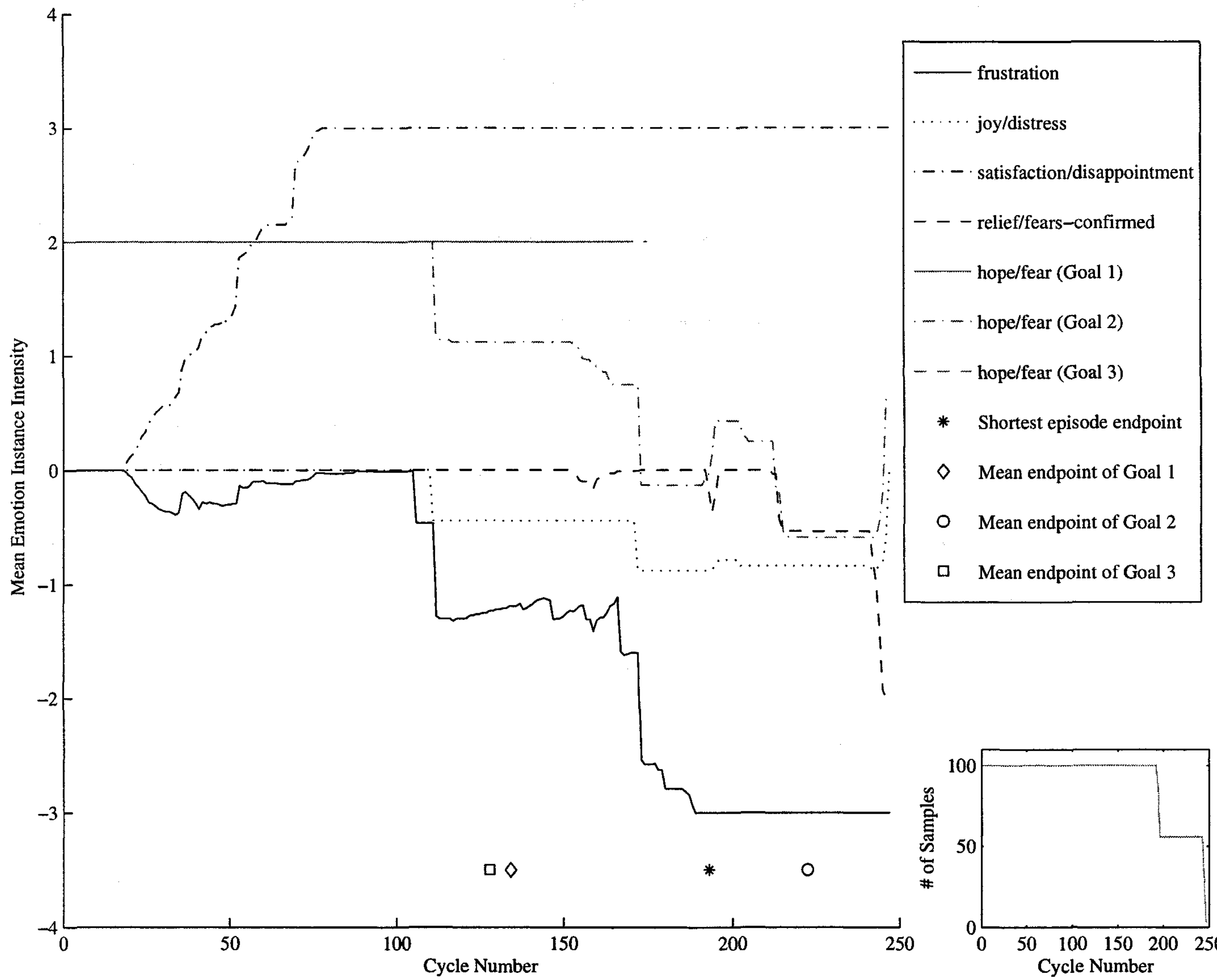

Figure A.59: Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

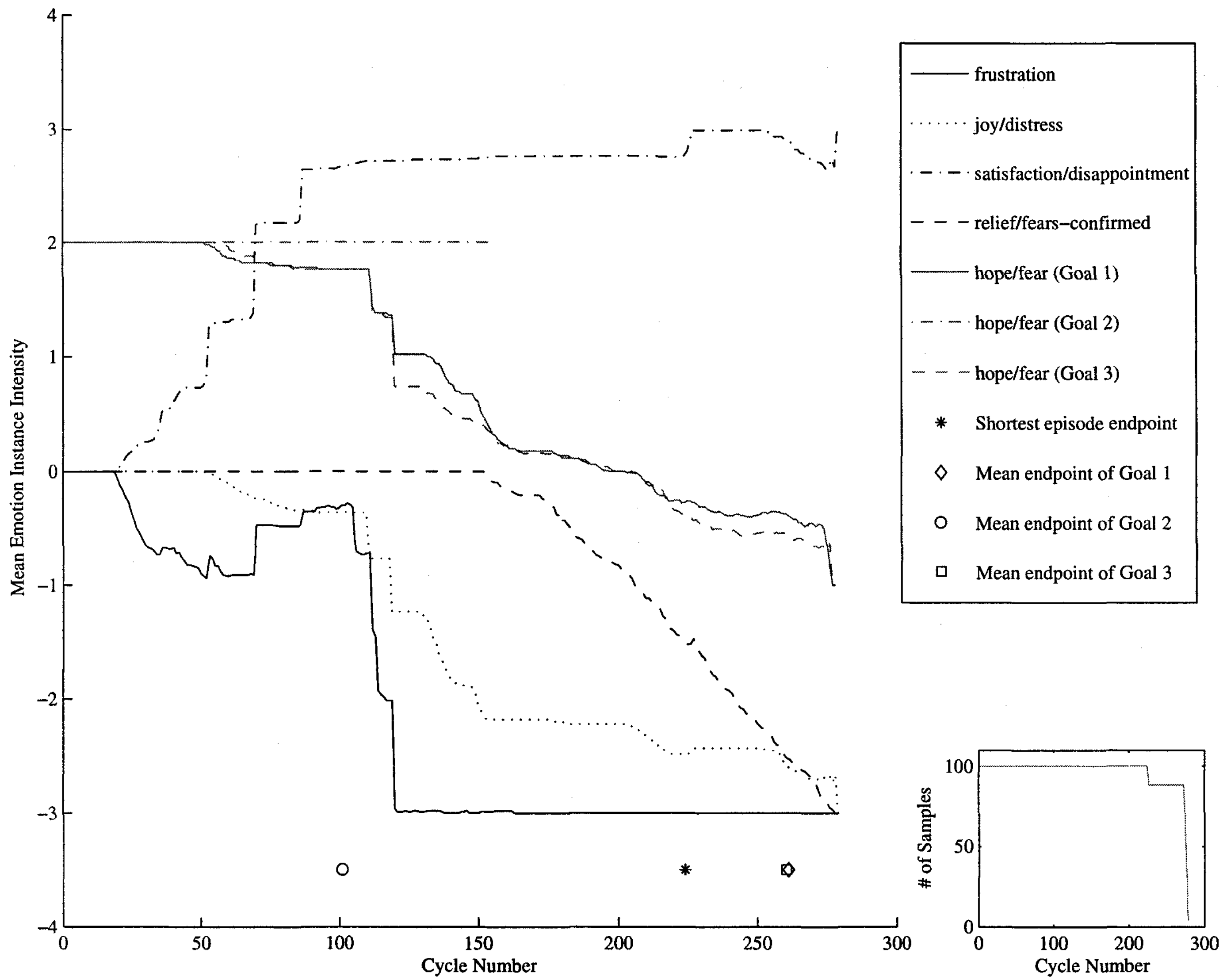

Figure A.60: Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

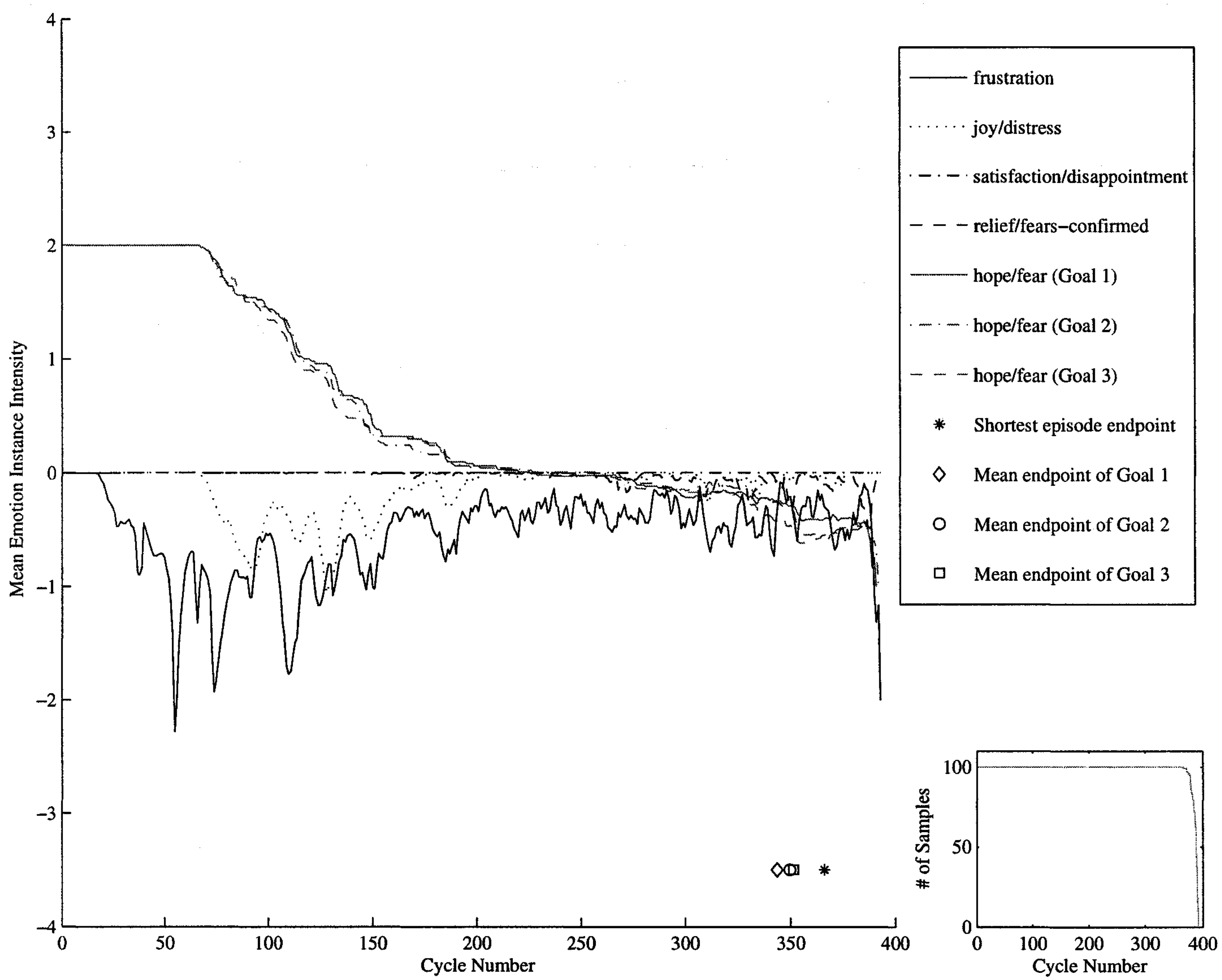

Figure A.61: Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

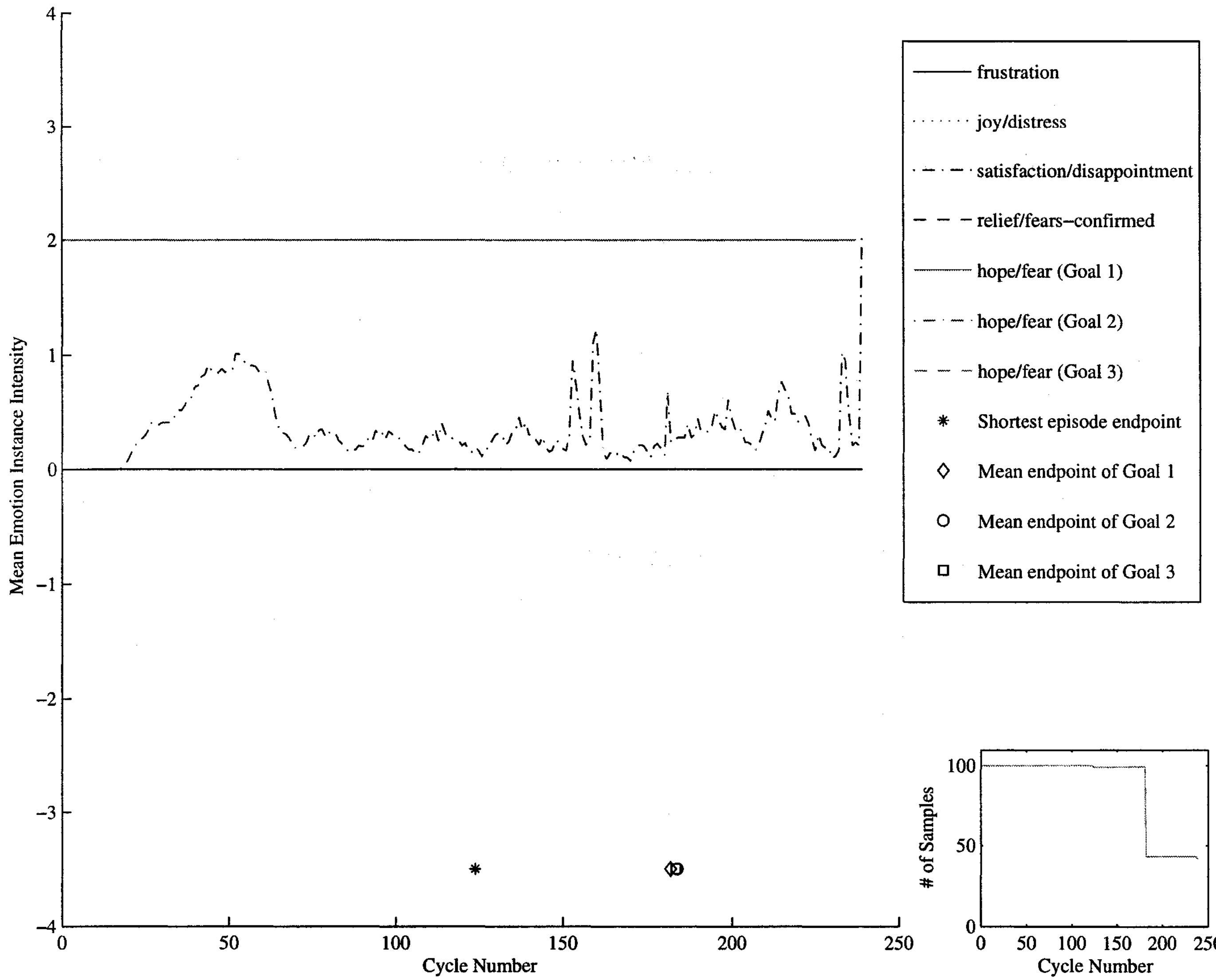

Figure A.62: Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience.

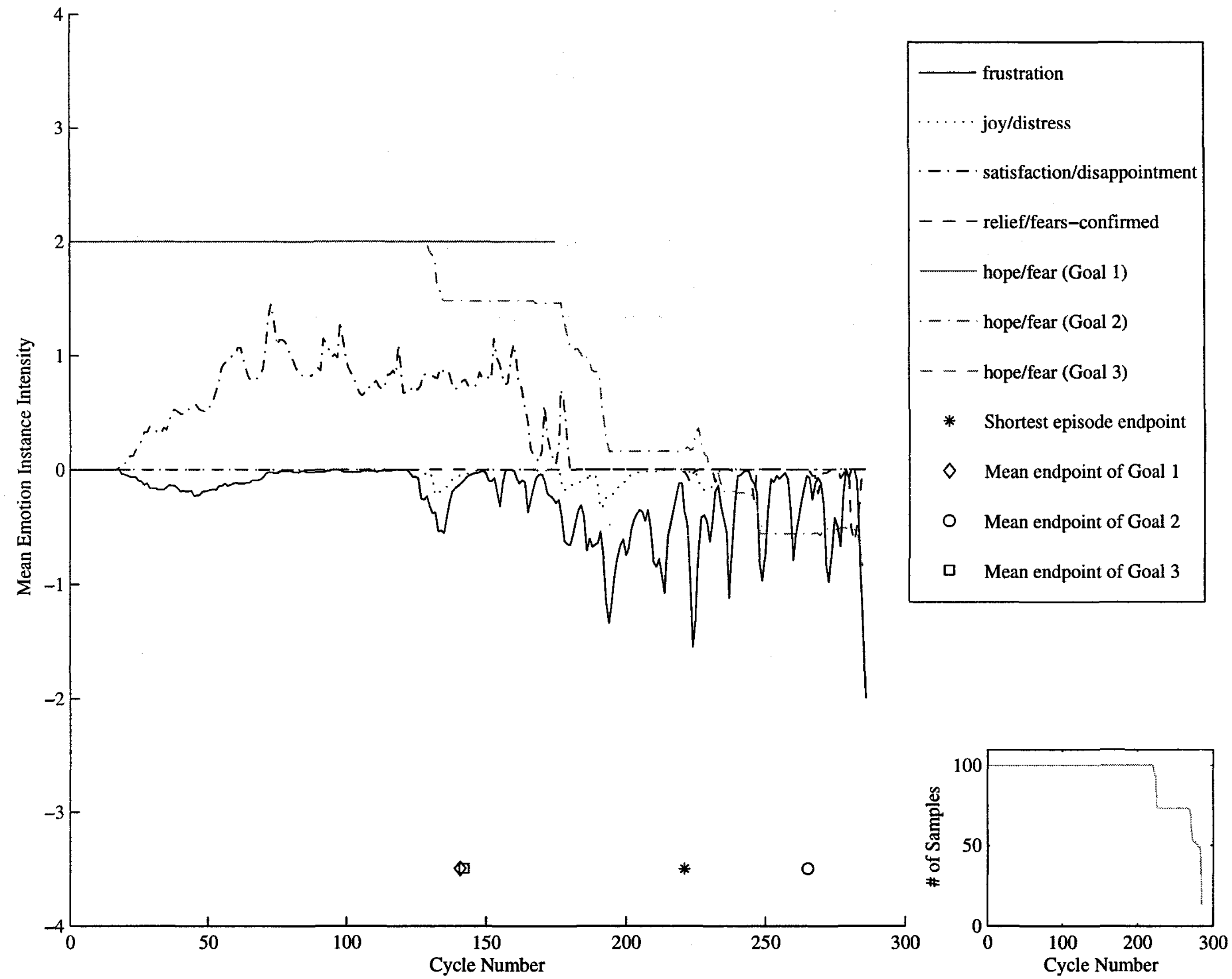

Figure A.63: Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience.

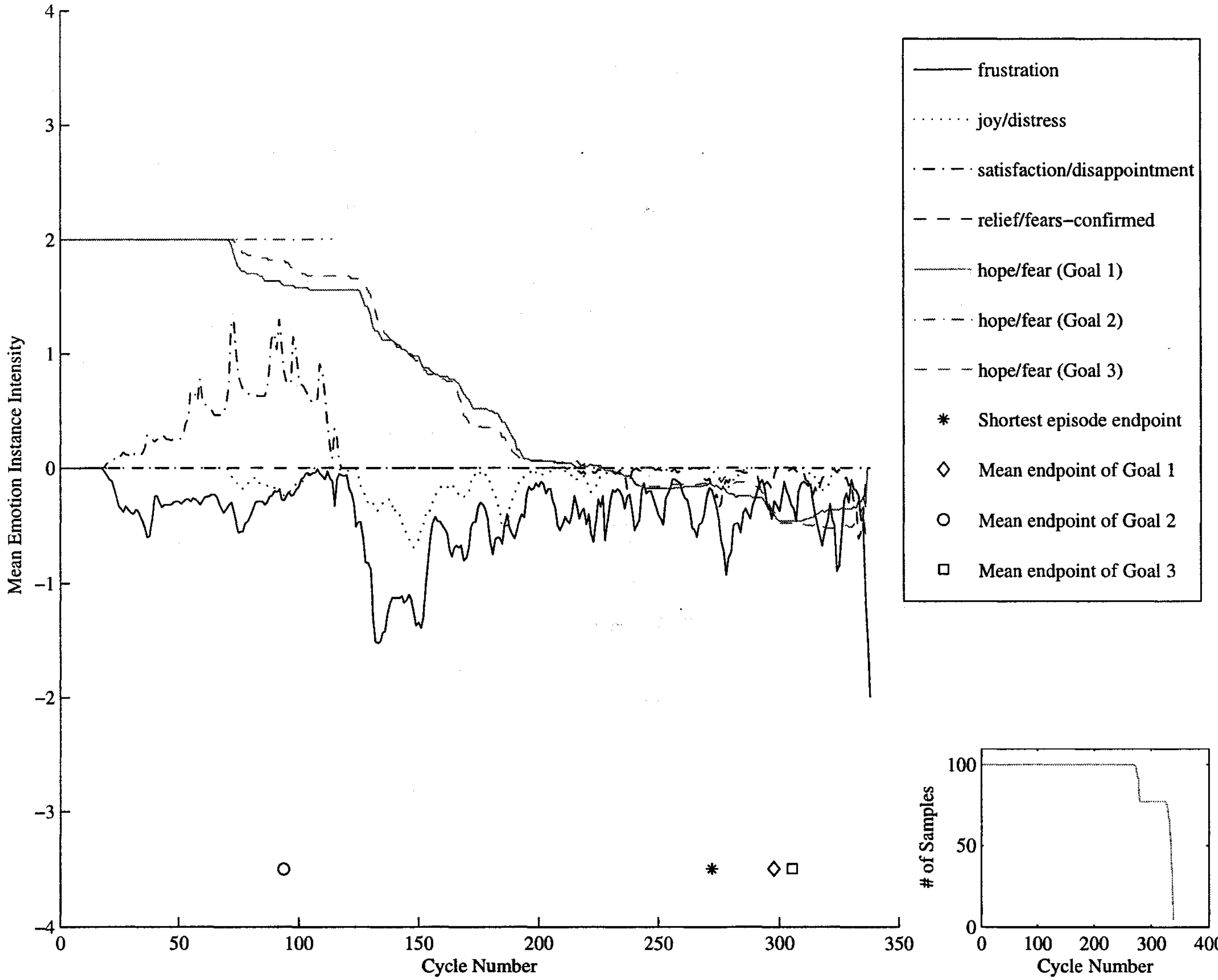

Figure A.64: Emotion chart for the noisy emotional agent in the asymmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

A.2.3 Persistence

For each test combination, we present mean persistence and normalized mean persistence. All test combinations use concurrent goals.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.63	5.80	5.31
Rational (noisy)	1.61	1.67	1.63
Emotional (non-noisy)	5.53	5.44	5.49
Emotional (noisy)	1.26	1.29	1.29

Table A.49: Mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	0.43	0.45	0.41
Rational (noisy)	0.12	0.13	0.13
Emotional (non-noisy)	0.43	0.42	0.42
Emotional (noisy)	0.10	0.10	0.10

Table A.50: Normalized mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	9.00	9.00	9.00
Rational (noisy)	1.52	1.52	1.46
Emotional (non-noisy)	5.59	5.42	5.45
Emotional (noisy)	2.55	2.53	2.52

Table A.51: Mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1.00	1.00	1.00
Rational (noisy)	0.26	0.26	0.26
Emotional (non-noisy)	0.62	0.60	0.61
Emotional (noisy)	0.32	0.32	0.30

Table A.52: Normalized mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	9.00	7.74	9.00
Rational (noisy)	1.58	3.05	1.64
Emotional (non-noisy)	5.00	6.47	5.00
Emotional (noisy)	2.40	4.04	2.55

Table A.53: Mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1.00	0.60	1.00
Rational (noisy)	0.25	0.23	0.27
Emotional (non-noisy)	0.56	0.50	0.56
Emotional (noisy)	0.32	0.31	0.32

Table A.54: Normalized mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	6.02	9.00	6.02
Rational (noisy)	1.79	1.73	1.83
Emotional (non-noisy)	4.78	4.84	4.74
Emotional (noisy)	1.82	3.03	1.79

Table A.55: Mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	0.46	1.00	0.46
Rational (noisy)	0.14	0.25	0.14
Emotional (non-noisy)	0.37	0.54	0.36
Emotional (noisy)	0.14	0.37	0.14

Table A.56: Normalized mean persistence for the asymmetric concurrent scenario with no E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.44	5.68	5.60
Rational (noisy)	1.65	1.62	1.64
Emotional (non-noisy)	4.00	3.90	3.81
Emotional (noisy)	1.90	1.87	1.85

Table A.57: Mean persistence for the asymmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	0.42	0.44	0.43
Rational (noisy)	0.13	0.12	0.13
Emotional (non-noisy)	0.31	0.30	0.29
Emotional (noisy)	0.15	0.14	0.14

Table A.58: Normalized mean persistence for the asymmetric concurrent scenario with E-Plans, with the Failure-Failure-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	9.00	9.00	9.00
Rational (noisy)	1.51	1.53	1.61
Emotional (non-noisy)	2.86	3.02	2.95
Emotional (noisy)	1.65	1.67	1.63

Table A.59: Mean persistence for the asymmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1.00	1.00	1.00
Rational (noisy)	0.26	0.25	0.25
Emotional (non-noisy)	0.32	0.34	0.33
Emotional (noisy)	0.21	0.22	0.20

Table A.60: Normalized mean persistence for the asymmetric concurrent scenario with E-Plans, with the Success-Success-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	9.00	7.51	9.00
Rational (noisy)	1.56	3.14	1.59
Emotional (non-noisy)	3.64	6.31	3.67
Emotional (noisy)	2.08	4.59	2.14

Table A.61: Mean persistence for the asymmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	1.00	0.58	1.00
Rational (noisy)	0.29	0.24	0.28
Emotional (non-noisy)	0.40	0.49	0.41
Emotional (noisy)	0.26	0.35	0.26

Table A.62: Normalized mean persistence for the asymmetric concurrent scenario with E-Plans, with the Success-Failure-Success problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	5.96	9.00	5.99
Rational (noisy)	1.81	1.57	1.79
Emotional (non-noisy)	3.57	4.57	3.39
Emotional (noisy)	2.16	3.22	2.21

Table A.63: Mean persistence for the asymmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

Agent Type	Goal 1	Goal 2	Goal 3
Rational (non-noisy)	0.46	1.00	0.46
Rational (noisy)	0.14	0.25	0.14
Emotional (non-noisy)	0.27	0.51	0.26
Emotional (noisy)	0.17	0.42	0.17

Table A.64: Normalized mean persistence for the asymmetric concurrent scenario with E-Plans, with the Failure-Success-Failure problem solving experience.

Appendix B

Narrative Experiment Materials

This appendix includes experimental materials developed for the narrative experiments discussed in Chapter 7. We include here the materials common for each subject: an instruction sheet, an introduction given before the narrative trace, a set of questions given after the narrative, and a debriefing sheet.

Instruction sheet:

Characters in Action Plays

The next pages present a portion of a small stage play, called "The Ward." It is written in a unusual modern style.

There is no dialogue. Instead, a playwright who follows this style writes from the point of view of the audience, describing how the stage setting looks, the actions that the character takes, and other features of the character that the audience can observe.

The entire script is a series of very simple stage directions describing the visuals of the scene, the character's decisions, and the character's movements.

In this study, we ask that you first read the play and then answer a set of questions about the character in it. This should take about 20 minutes.

You may turn the page and begin.

If you have any questions, raise your hand and ask the experimenter.

Introduction (page 1 of 2):

The Ward

It is a sunny day. In the background we see a farmhouse.

The left, from our vantage point, shows the view of a cornfield, partially obscured by a small shed. The right, from our vantage point, is formed by the view of a large beet field. Birds are circling above both fields.

To the right of the farmhouse door, from our vantage point, in front of a window, we notice a wooden block with a hatchet in it; or rather, a large piece of wood is lying on the block, which is standing at an angle, and a hatchet is sticking in the piece of wood.
Round about the chopping block we notice many pieces of chopped wood, and also, of course, chips and splinters, strewn about the ground.

Already, upon first glance, we have seen someone sitting next to the chopping block, on a stool: a figure. Now, after having briefly taken in the other features of the scene, we turn back to the figure sitting on a stool in the sunshine in front of the house.

He — the figure is that of a male — is dressed in rural garb: that is, he is wearing overalls over his trousers; his shoes are heavy; on top, the person is only wearing an undershirt. No tattoos are visible on his arms. The person wears no covering on his head. This is the ward.

The inside of the house is a single room, with a dark bare bulb hanging from the rafters. A newspaper lies in the crack of the door.

We see a large block calendar hanging on the right wall of the room. A rather large table and two chairs sit in the middle of the room. Distributed through the room are an assortment of other objects.

We know that the ward has been assigned a number of tasks to complete this afternoon. The ward knows this, too.

The next task he needs to accomplish is replacing the burnt out light bulb in the house. We know that there

Introduction (page 2 of 2):

are a few ways he could achieve this task. He could use a long, light bulb grabbing device to reach the bulb, or climb the ladder, or climb the furniture. Some of these plans could be easier for the ward to try than others. However, we know that he believes some of them are more likely to work out than others.

We also know some things that the ward does not know. For instance, we know that a new light bulb can be found on a shelf in the house, and the ladder is out in the beet field. The light bulb grabber is stored in the shed. However, both the ladder and the chairs are wobbly and not good for climbing on.

The ward stands up.

Questions (page 1 of 4):

Below and on the next few pages, there are questions about your reaction to the character, "the ward," from the short play you just read.

For the first set of questions, indicate your answer by circling a number on the rating scale for each question. Please add any comments that explain your answer.

Do you like the ward? *circle a number*

1 --------- 2 --------- 3 --------- 4 --------- 5 --------- 6 --------- 7
not at all neutral very much

Why?

(continued next page...)

Questions (page 2 of 4):

Do you want him to succeed? *circle a number*

1 --------- 2 --------- 3 --------- 4 --------- 5 --------- 6 --------- 7
not at all neutral very much

Why?

Do you understand him? *circle a number*

1 --------- 2 --------- 3 --------- 4 --------- 5 --------- 6 --------- 7
not at all neutral very much

Why?

Do you identify with him? *circle a number*

1 --------- 2 --------- 3 --------- 4 --------- 5 --------- 6 --------- 7
not at all neutral very much

Why?

(continued next page...)

Do you approve of him? *circle a number*

1 --------- 2 --------- 3 --------- 4 --------- 5 --------- 6 --------- 7
not at all neutral very much

Why?

Questions (page 3 of 4):

Do his actions make sense to you? *circle a number*

1 --------- 2 --------- 3 --------- 4 --------- 5 --------- 6 --------- 7
not at all neutral very much

Why?

Are you on his side? *circle a number*

1 --------- 2 --------- 3 --------- 4 --------- 5 --------- 6 --------- 7
not at all neutral very much

Why?

Do you think he has a future? *circle a number*

1 --------- 2 --------- 3 --------- 4 --------- 5 --------- 6 --------- 7
not at all neutral very much

Why?

(continued next page...)

Below is a list of paired adjectives. For each pair, circle the one that you think is the *better* description of the ward.

1.	nice	not nice
2.	crazy	rational
3.	satisfied	unsatisfied
4.	discontented	happy
5.	satisfied	disappointed

Questions (page 4 of 4):

6.	disappointed	relieved
7.	serious	playful
8.	relaxed	frustrated
9.	pessimistic	optimistic
10.	stable	flighty
11.	hardworking	lazy
12.	fearful	hopeful
13.	accepting	intolerant

This is the end of the questions. Please give this booklet to the experimenter. Thank you for your help.

Debriefing sheet:

Debriefing Form for *Characters in Action Plays*

The study that you participated in today is one example of research that applies theories of human emotion to models of artificial intelligence. From a psychology perspective, these models help us to examine our theories and assumptions underlying the functions of emotion in different contexts. From an artificial intelligence perspective, the study of such models helps us to develop computer applications than can interact with humans in realistic and intelligent ways to solve real world problems. Examples might include intelligent tutor programs that help us learn, expert systems that assist doctors in performing emergency surgeries, or intelligent agents that help us to pay bills or reserve airline or concert tickets as just a few examples.

In this study we are interested in determining which settings in the computer program generate characters that are the most believable. The play you read was generated by a computer program and based on a real play called "My Foot, My Tutor", written by German playwright Peter Handke. In this style, emotions and inner states are conveyed by simple actions.

We are interested in determining which settings in the computer program generate characters that are the most believable.

The computer program has a representation of knowledge about how to solve tasks in a simple, simulated world. The emotional element is modelled as a set of emotion dimensions (e.g., distress/joy, relaxed/frustrated, fearful/hopeful). The main feature of the system that impacts these emotion dimensions is the success or failure of plans that the character attempts in the world. The other elements that affect these emotion dimensions include some initial personality settings (how easily a character becomes frustrated through failure). These emotions impact the character's subjective judgements about a plan's difficulty and its probability of success, and how important a goal is to him. This, in turn, determines the choices the character makes within the simulated world.

The system simulates a world, events in the world, and the character within that world. It updates the simulated world after each action the character task, and then updates the emotional state of the character depending on the factors outlined above. The character's next choice about what to do in the world is affected by this emotional state.

This study is not about assessing your emotional state or any other features about you. Instead, we are using you and your answers to judge what settings in the software generate the most believable character. Different settings were used, and each setting combination generates different behaviour in this simple world. The results will help us validate the underlying mechanisms for creating characters that seem 'believable'.

There is a vast literature on computer models of emotion in both cognitive psychology and in artificial intelligence, and if this is interesting to you, you can easily references to this on the web and citation databases.

Thank you for participating.