University of Alberta

Real-Time Optimization of Gasoline Blending with Uncertain Parameters

By

Dayadeep S. Monder ©

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

in

Process Control.

Department of Chemical and Materials Engineering

Edmonton, Alberta

Spring 2001

0-612-60472-1

University of Alberta

Library Release Form

Name of Author: Dayadeep S. Monder

Title of Thesis: Real-Time Optimization of Gasoline Blending with Uncertain Parameters

Degree: Master of Science

Year this Degree Granted: 2001

[signature]

196, Sector 71
Chandigarh, India 160059

Date: November 3rd, 2000

"In these matters, the only certainty is that nothing is certain."
Pliny the Elder

"Prediction is difficult, especially of the future..."
Niels Bohr

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Real-Time Optimization of Gasoline Blending with Uncertain Parameters** submitted by **Dayadeep S. Monder** in partial fulfillment of the requirements for the degree of **Master of Science** in **Process Control**.

J. Fraser Forbes (Supervisor)

Sirish L. Shah

Tongwen Chen

Date: November 3rd, 2000

To my parents

Acknowledgements

Foremost, I would like to thank my parents for all that they have given to me, and for granting me the freedom to explore on my own. For helping me through this work and for his extraordinary support, patience, and sage advice, I am grateful to my supervisor, Dr. J. Fraser Forbes.

I believe that I have been blessed with some of the best friends a person can hope for. My eternal gratitude goes out to the ones that were with me, in thought and in person, during my stay at U of A. From my good buddies from back home: Ashish, Gagan, Harmeet, Samiksha, Aashish, Sachin; to the new ones I made in Edmonton: Rohit, Arun, Indra, Sonia, Gurinder, Al (you are sorely missed, dude), Andru, Nikhil, Laksh, and Aseema.

I would like to thank the entire CPC Group at U of A for welcoming me into their midst and for tolerating me as long as they have. I would also like to express my gratitude to people at my current place of employment: Mark, Ling, Julie, Dave, Benny and James, for their support and encouragement during the final struggle. To all the people I have missed here, borrowing from one of my favourite bands: "You know who you are. Take a bow."

Abstract

Gasoline is a key revenue generator for the petroleum refining industry. Typically, blend recipes are determined such that property specifications (*e.g.* Octane Number, vapour pressure, *etc.*) are met while maximizing the profitability of the blend. Lower level process controllers enforce these blend recipes, ensuring that the property specifications are met and quality give-away is minimized, while compensating for disturbances and plant/model mismatch.

None of the current blending Real-Time Optimization (RTO) approaches treat uncertainty in the constraint parameters (feedstock qualities, *etc.*) directly. As shown by Singh *et al.* [1996], varying constraint parameters can cause the calculation of infeasible solutions, if the expected values of the parameters are used. In this work, methods are proposed to explicitly include constraint parameter uncertainty in the dynamic formulations of the Gasoline Blending Optimization problem. These new formulations borrow from a branch of Stochastic Programming (SP) called Probabilistic Programming (PP). The performance of the proposed SP based optimizers is compared with that of conventional blending optimizers. This thesis thus provides a new approach to solving the Gasoline Blending Optimization problem, which explicitly includes uncertainty information in the problem formulation.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Optimization is an integral part of engineering and is used by engineers (both heuristically and mathematically) to design new processes/equipment, "improve" existing processes, and to optimize process operations [Edgar and Himmelblau, 1988]. Process Operations Optimization is essential if a manufacturer wants to stay competitive in the industry [White and Hall, 1992]. The objective of Process Operations Optimization is to determine an operations policy that optimizes the economic benefit of producing the final product. In most cases, the economically optimal policy is calculated using steady-state optimization of the plant operations [Forbes, 1994].

Most steady-state optimization methods can be divided into: 1) direct search based methods; 2) model-based methods [Garcia and Morari, 1981]. Direct search methods use on-line plant experimentation to perturb the plant to different operating regions and measure the performance index (cost and/or profit for economic optimization) in these regions. Thus, given enough experimentation, a profit surface can be mapped over the different possible operating regions. Most direct methods use an iterative policy (continuous experimentation) to refine this profit surface and try always to move the plant to an operating region where the process performance index improves. More recent developments to the direct search methods have been in identifying dynamic models for the plant from process perturbation and using these to determine economically optimal steady-state moves for the process (*e.g.*, McFarlane and Bacon, 1989). However, there are some fundamental limitations to the use of all direct search methods. The amount of plant experimentation involved can be prohibitive for large scale plants with many degrees of freedom. Moreover, large time constants in these complex interconnected units mean that it takes a long time for the units to come to steady state after each experiment. Thus, the sheer number of experiments that need to be performed, and the amount of time required by each experiment, combine to make direct search based methods particularly difficult to use for large scale optimization problems.

Model-based optimization methods represent the process with a steady-state or dynamic mathematical model and use the model to predict optimal operation. Thus, instead of working directly on the process, these methods use a mathematical approximation of the process. Optimization algorithms are then used to optimize the model and the solution is implemented on the plant (after applying certain checks).

Most large scale units in the chemical and petrochemical industry are time-varying in nature. This means that the underlying processes are changing all the time. This could be due to various reasons: heat exchanger fouling, reactor catalyst decay, feedstock compositions changes, and so forth. For a time-varying process, the economically optimal operation conditions are a trajectory in time that the plant must follow. Process Optimization structures which track these changes in the process and solve the optimization problem for the optimal operations policy are known as Real-Time Optimization (RTO) systems or On-line Optimizers. For large-scale industrial implementations, the size of the model and the resulting computational load currently prohibits the use of dynamic process models to calculate the optimal trajectory in real-time. As a result, most RTO systems are steady-state, model-based optimization systems which solve the steady-state optimization problem for the process, recursively. Thus, the model representing the process must be updated and optimized at a frequency, which (ideally) keeps pace with the process changes. The process models themselves are an approximation of the physical plant, and thus, have both structural as well as parametric uncertainty associated with them. This uncertainty in model structure and parameters needs to be taken into account in the design of RTO systems.

Gasoline is a key revenue generator for the petroleum refining industry [DeWitt *et al.*, 1989]. Gasoline blending operations is one of the petroleum refining processes where RTO systems are used fairly extensively [DeWitt *et al.*, 1989; Sullivan, 1990; Ramsey and Truesdale, 1990; Bain *et al.*, 1993]. The motivation behind using an RTO system for gasoline blending optimization, is to address the variation in feedstock qualities and market price fluctuations. Gasoline blending is one of the few processes that are fairly simple to model (essentially no model dynamics) along with being industrially relevant. Also, the blending optimization problem has a relatively low number (typically < 100) of decision variables. This makes gasoline blending optimization a good candidate to do RTO studies on. Conventional gasoline blend optimizers do not take uncertainty in the optimization problem parameters into account while solving for the optimal blend recipes. In this thesis, the issue of uncertainty in blending optimization parameters is considered and solutions are offered for the uncertain feedstock qualities case.

1.1 Real-Time Optimization (RTO)

The structure for a typical steady-state closed loop Real-Time Optimization (CLRTO) system is shown in Figure 1.1. The various RTO sub-systems or components: Measurement, Data Validation, Model Updating, Optimization, Command Conditioning (also called Results Analysis), and Control, are described below.

1.1.1 RTO Components

The **Measurements** block is the set of sensors used to monitor the process. Process measurements are used in turn to update the process model in the optimizer, after they have been checked for validity.

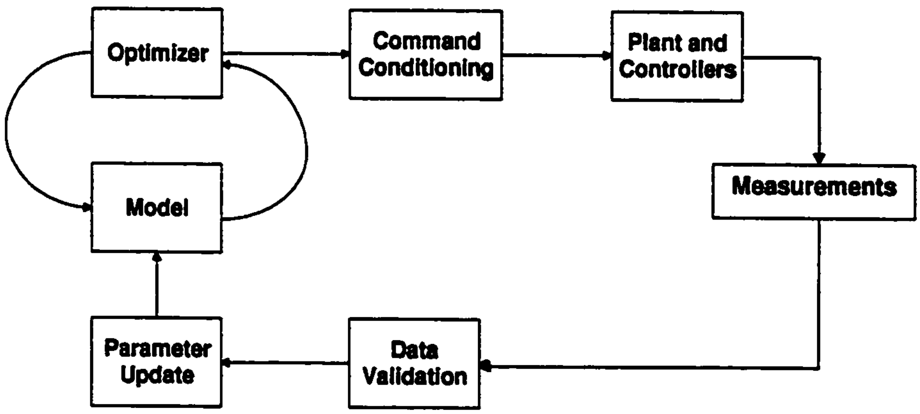

Figure 1.1: RTO Components and Structure [Fraleigh, 1998]

The data from the measurement set is examined for several purposes in the **Data Validation** sub-system. Within this sub-system, process measurements are first monitored for the detection of steady-state in the process. The other sub-processes in the Data Validation block include gross error detection, and data reconciliation. The purpose of this sub-system is to ensure that "reliable" data is used in the subsequent RTO sub-systems.

The validated, steady-state measurements from the process are used to update the process model used for optimization. The **model updating** usually takes the form of calculation of certain parameters (*e.g.*, heat transfer coefficients, distillation tray efficiencies, *etc.*) in the optimization model. This sub-system is responsible for keeping the model current so that process changes are tracked effectively.

The **Optimizer** block is where the steady-state optimization problem, using the updated model, is solved to generate a feasible operating point that maximizes profit (or minimizes cost of operation) at the current time.

The Optimizer generates setpoints defining the model-based optimal operating point. These setpoint estimates, however, are uncertain due to error propagation through the RTO loop. **Results Analysis** is the process of examining the optimizer outputs to ensure that the new setpoint estimates actually represent a significant change in the process before the setpoints are implemented.

The final sub-system in the RTO loop, the **Control** sub-system, takes the optimizer output in the form of setpoints and implements them on the process to take it to the newly calculated optimum.

1.1.2 Design Issues in RTO

As the RTO system is a closed loop system (Figure 1.1), each of the above mentioned sub-systems affects the overall performance of a Real-Time Optimizer. Thus, each component in the loop has to be carefully designed such that the overall process of on-line optimization is successful. The Measurement sub-system is where the overall CLRTO process begins. The selection of measurements to be used as inputs for the RTO system is very important considering the process model updates (and subsequently the optimization step itself) are dependent upon current information about the plant. Krishnan *et al.* [1992] propose measurement selection based on minimizing errors in estimation of model parameters (parameters in the process model which are updated by the RTO system to reflect changes in plant conditions). This approach, however, only considers the effects of measurement selection on the Model Updating sub-system. In another proposed measurement selection strategy, Fraleigh [1998] bases the selection criteria on overall performance of the RTO system. In the first stage of this approach, viable measurement sensor sets are identified subject to observability criteria for the parameters to be updated. In the second stage, process setpoint quality and expected profitability (derived using Design of Experiments theory) are combined into a Sensor System Design Cost (SSDC) criterion, which can be used to compare possible measurement sensor sets. The SSDC criterion is also useful in determining the optimal placement of additional sensors in the process to increase the RTO system's performance.

The Data Validation block takes raw plant data and performs a series of statistical tests on it. The first of these is usually verification that the plant/process being optimized is at steady state. Steady-state detection [Cao and Rhinehart, 1995] is necessary before the measurements can be used for updating the steady-state models. If the plant is experiencing transients, the RTO system usually goes into a suspended state until the transients die down. At least one commercial implementation of on-line optimizers, MDC Technology's RTO+ [Hyprotech Ltd., viewed 24 September, 2000] does not employ steady state detection as part of the Data Validation block.

The other essential steps before the steady-state measurements can be used in model updating are Gross Error Detection [Crowe, 1988; Tong and Crowe, 1997, Kelly, 1999], and Data Reconciliation [Tjoa and Biegler, 1990, Crowe, 1996]. Gross Error Detection involves identifying and removing gross errors (*e.g.*, readings from failed sensors) while Data Reconciliation tries to ensure that the corrected measurements satisfy the material and energy balances around and inside the process. Both the above sub-processes depend heavily on redundancy in the input process measurements to isolate faulty measurements and reconcile current plant information passed on to the Model Updating sub-system.

Selection of appropriate models to represent the plant in model based on-line optimization systems is one of the first RTO design decisions made. It was shown by Biegler *et al.* [1985], that a model is adequate for optimization if gradients of the model's objective function with respect to the decision variables can be made to match those of the plant. This requirement was relaxed by Forbes *et al.* [1994] by showing that only the reduced gradients of the objective function [Fletcher, 1987] need to be matched to those of the plant at the optimum. Model fidelity in the context of an RTO system is dependent upon both the model structure as well as the selection of parameters to be updated. Model adequacy requirements that take these into consideration for a closed loop RTO system are discussed by Forbes [1994]. The problem of selecting suitable parameters to be updated in the model(s) used has also been looked at by researchers. Updating parameters that have the most effect on the profit function and/or change the active constraint set is suggested by Krishnan *et al.* [1992]. An alternate, more direct approach [Forbes, 1994], is to use Model Adequacy conditions to choose updated model parameters. That is, if a set of parameters exists which can be manipulated such that the model optimum coincides with the plant optimum, this set of parameters should be updated online. Another important model parameter updating issue is the observability [Krishnan, 1990] and the degree of observability [Singh, 1997] of the updated parameters from the selected measurements set. The choice of the solution algorithm used in the actual parameter estimation also plays a major role in RTO performance. A robust estimation scheme is essential due to the nature of the data (a single set of data corrupted with sensor errors) being operated upon. Fraleigh [1998] proposes the use of more robust least-squares based solution algorithms instead of the conventional (less computationally expensive) back-substitution algorithms.

The core sub-system in the RTO loop is the optimizer itself. Two important factors influencing the choice of an algorithm in the optimizer sub-system are: *a*) ability to incorporate the process model chosen for the plant into the formulation of the optimization problem, *b*) computational tractability *i.e.*, the ability to solve the optimization problem posed in a reasonable amount of time. Optimization theory / Operations Research is an important branch in mathematics that has been studied extensively both by theoretical mathematicians and by engineers *e.g.*, Edgar and Himmelblau [1988]; Gill *et al.* [1981]; Avriel [1976]; Wright [1997].

Before the setpoints generated by solving the process optimization problem can be implemented by the control system in the plant, the optimizer solution goes through the Command Conditioning or Results Analysis sub-system. This post optimality statistical analysis is done to check if the process moves dictated by the optimizer solution are in fact worthwhile. The optimizer results are implemented if the statistical tests suggest that the difference in the optimum of the plant and the model is significant and is not due to assorted errors propagated through the RTO loop [Miletic and Marlin, 1998; Zhang *et al.*, 2000].

Although each RTO sub-system is designed to meet specific requirements for itself and sub-systems dependent upon it, selection of an RTO design to be implemented should ultimately be dependent on expected performance of the RTO system. One of the first design approaches using this philosophy was the Design Cost approach by Forbes and Marlin [1993, 1996] to model-based optimization system design. The Design Cost method evaluates the cost (profit lost with respect to perfect optimization) of an RTO design by combining the costs of setpoint bias and variance when implemented. This method was enhanced (Extended Design Cost or EDC) by Zhang and Forbes [2000] to include the cost incurred while the process is in transition from one process optimum to another. Another approach that considers the complete RTO loop in making RTO design decisions is the method of average deviation from optimum proposed by de Hennin [1994] and extended by Loeblein [1997]. This approach consists of mapping the different error sources (measurement errors, modelling errors, *etc.*) in an RTO loop through the parameter update and optimization sub-systems to examine their effect on the performance of an RTO design. An analytical expression is derived for the average deviation from the true optimum for an on-line optimizer by integrating the deviation from true optimum over the various error sources. This average deviation from optimum is then computed and compared for the available RTO designs to select the most profitable scheme.

In this section, on-line or real time optimization was introduced. In the next section of this chapter, the process of primary focus in this study, gasoline blending, is introduced.

1.2 Gasoline Blending

Gasoline is produced by blending different streams from various upstream processes. Any grade of gasoline should be in the correct boiling range (100 to 400^oF, [Gary and Handwerk, 1994]). This requirement restricts the choice of streams (boiling fraction cuts) that can be used as gasoline blend feedstocks (Figure 1.2). Despite this restriction, some refineries can have up to 20 different gasoline blending feedstocks [Givens, 1985]. The main feedstocks used are:

1. Light Straight Run naphtha or LSR, which is the gasoline boiling range cut from the atmospheric distillation tower.
2. Catalytic cracker gasoline (*i.e.*, the gasoline cut from the fluidized catalytic cracking unit or FCCU).
3. Reformate, the gasoline from the catalytic reforming unit.
4. Alkylate, the gasoline cut from the liquid catalyzed alkylation unit.
5. n-Butane, normal butane from various processes including the crude atmospheric tower, FCCU, *etc.*
6. 'Hydrocrackate', the gasoline fraction from the hydrocracker.

In this work only the first five will be considered as the feedstocks available.

Each grade of gasoline has to have certain properties in order to meet specifications. Some of the most important ones are described here.

Octane Number is an indicator of the fuel's anti-knock properties. There are two measures generally used: Research Octane Number (RON), and Motor Octane Number (MON). The higher the octane number, the better the gasoline's anti-knock capability and the smoother the engine runs.

Reid Vapor Pressure (RVP) is a measure of the volatility of the fuel. The gasoline has to be volatile enough to vaporize on injection into the combustion chamber. At the same time it being excessively volatile could lead to vapor-lock in the fuel

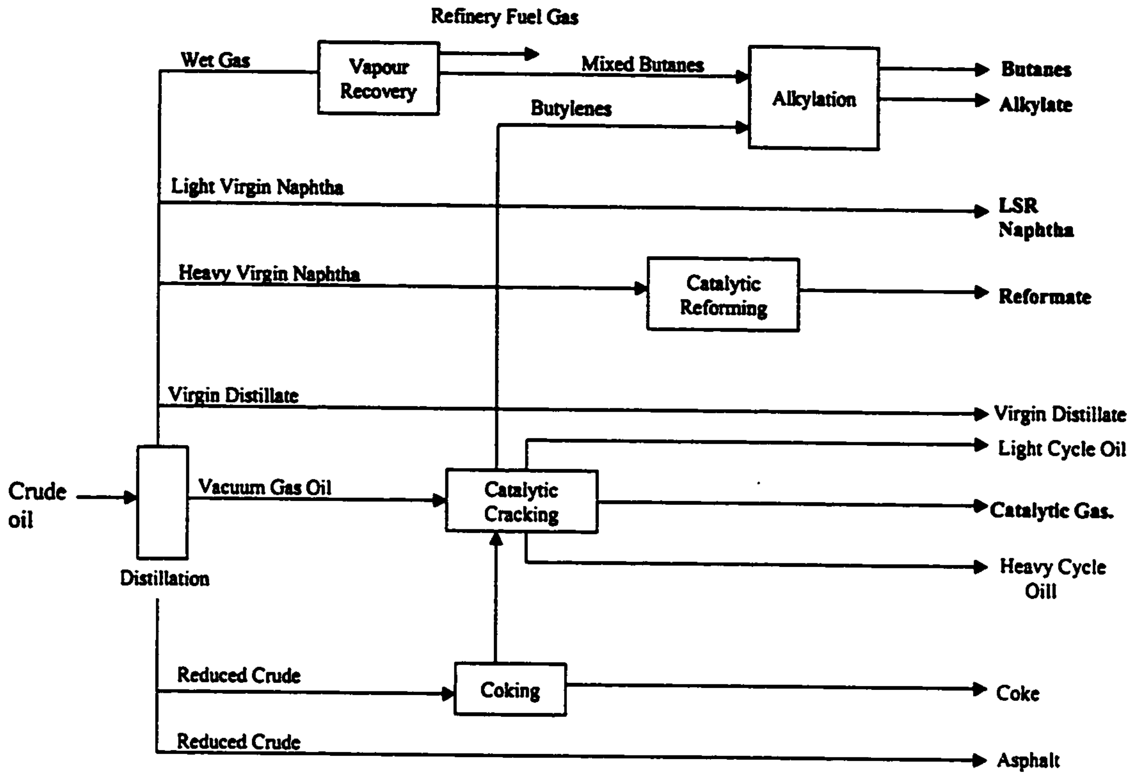

Figure 1.2: Flow Diagram for a typical Refinery (gasoline feedstocks in bold) [Singh, 1997]

line. Environmental regulations also limit the maximum RVP to reduce emissions of volatile organic compounds (VOC).

Gasoline is a mixture of hydrocarbons with a range of different boiling points. To better characterize gasoline volatility, a standard laboratory batch distillation is performed. During this batch distillation, amount distilled (percent volume basis) to a given temperature is measured. Using several points along the temperature scale, the **ASTM Distillation** curve, is generated. This test is known as the ASTM D-86.

Total **Aromatics Content** (benzene, toluene, *etc.*), and **Sulphur Content** in motor gasoline is also regulated due to environmental concerns (vehicular emissions).

Although this thesis considers only the first three gasoline qualities mentioned above (*RON*, *MON* and *RVP*), for illustrative purposes, the methods presented here are general and can be easily extended to include all specifications on blended gasolines.

Issues in gasoline blending and real-time optimization were considered separately in the previous sections. In the next section, RTO techniques and formulations used in on-line optimization of gasoline blending are described.

1.3 RTO in Gasoline Blending

The huge volume of gasoline production implies that a small saving in the cost of production per unit can lead to a large increase in the total profit [Grosdidier, 1997]. Also, any benefits from improved control/optimization of processes upstream will be useless if the final blending step is "sub-optimal". These facts make on-line optimization of gasoline blending a very attractive proposition.

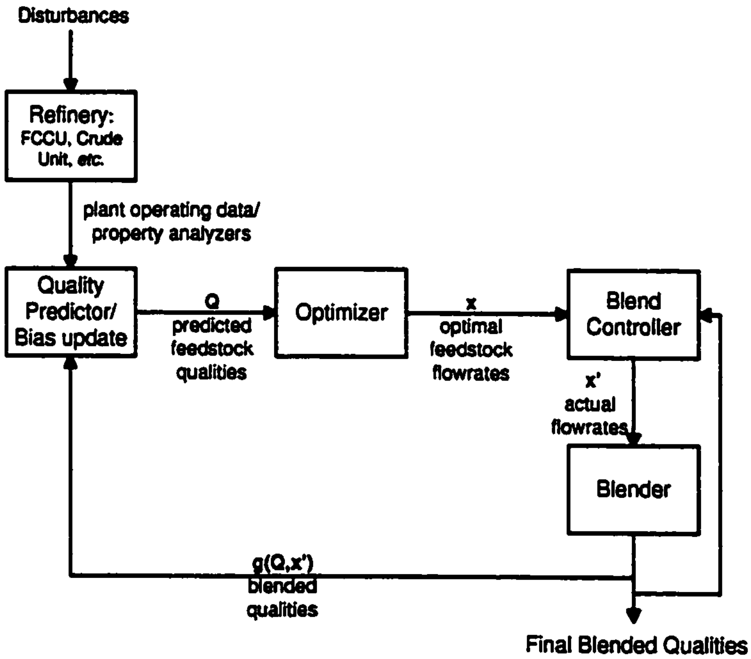

Figure 1.3: Gasoline Blending RTO Structure

In this section, the various conventional gasoline blending RTO structures in current use are described briefly. The distinguishing features and shortcomings of these optimizers are discussed. Some of the commercially available blend controllers and optimizers are also discussed briefly.

The general blending optimization problem (Problem 1.1 below) consists of maximizing an economic measure (usually profit) while ensuring that all the property specification, product demand and feedstock availability constraints are met.

$$
\begin{array}{ll}
\max\limits_{\mathbf{x}} & profit(\mathbf{c},\mathbf{x}) \\
\textit{subject to:} & \\
& \textit{blend properties}(\mathbf{Q},\mathbf{x}) \geq \textit{specifications} \\
& \textit{product produced}(\mathbf{x}) \geq \textit{product demand} \\
& \textit{feedstock used}(\mathbf{x}) \leq \textit{feedstock availability}
\end{array}
\tag{1.1}
$$

where: $\mathbf{c}$ is the vector containing price information for the feedstocks and the final blends, $\mathbf{x}$ is the flow-rate vector of the blend feedstocks and $\mathbf{Q}$ is the matrix containing property information for the feedstocks.

1.3.1 LP + Bias

The simplest RTO structure used in gasoline blending is a variation on the Linear Programming (LP) approach. A Linear Program (Gill *et al.*, 1981) is one of the simplest mathematical optimization formulations where both the objective function and the constraints are linear in the decision parameters ($\mathbf{x}$ in Problem 1.2). Known as the *LP + bias* (Linear Programming with bias updating) form [Singh, 1997], this formulation reduces the gasoline blending optimization problem to the following Linear Program, which is solved at each RTO interval:

$$
\begin{array}{ll}
\max\limits_{\mathbf{x}} & p\mathbf{1}^T\mathbf{x} - \mathbf{c}^T\mathbf{x} \\
\textit{subject to:} & \\
& \widetilde{\mathbf{Q}}\mathbf{x} \leq (\mathbf{1}^T\mathbf{x})\mathbf{s} - \boldsymbol{\beta} \\
& \mathbf{H}\mathbf{x} \leq \boldsymbol{\rho}
\end{array}
\tag{1.2}
$$

where: p is the selling price of the blended gasoline, $\mathbf{c}$ is the vector of cost prices of feedstocks, $\mathbf{x}$ the flow-rate vector of feedstocks, $\widetilde{\mathbf{Q}}$ the quality matrix containing the feedstock quality blending indices, $\mathbf{s}$ is the vector containing blend quality specifications, $\boldsymbol{\beta}$ the vector of bias terms used to update the blending model to maintain its accuracy, $\mathbf{H}$ is the matrix for demand and availability constraints, $\boldsymbol{\rho}$ the vector of demands and availabilities, and $\mathbf{1}$ is a column vector of ones, $\begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}^T$.

All the properties considered above (*RON*, *MON*, *RVP*) blend non-linearly (*i.e.*, the final *RON* of the blend will not usually be the volumetric average of the *RON*s of the feedstocks). There are many empirical and semi-empirical non-linear models available in literature ([Healy *et al.*, 1959] , [Stewart 1959*a,b*] , [Morris *et al.*, 1994] , [Rusin *et al.*, 1981] , [Vazques-Esparragoza *et al.* 1992], *etc.*) which describe property blending, but they cannot be used if the problem is to be formulated as a linear program. As a result, simpler, linear blending models are used in which the actual properties are transformed into blending indices whose volumetric average gives us the blended properties. These transformed qualities: blending *RON*, blending *MON*, and Reid Vapor Pressure Blending Indices are contained in the matrix $\widetilde{\mathbf{Q}}$. The bias vector, $\boldsymbol{\beta}$ is the difference between the actual qualities and the predicted qualities at the previous RTO interval:

$$\boldsymbol{\beta} = (\text{measured blend qualities} \;-\; \text{predicted blend qualities}) \times (1^T \mathbf{x}_{k-1}) \qquad (1.3)$$

The bias updating structure is one of the simplest model updating strategies used in Model Predictive Control (MPC) structures [Garcia *et al.*, 1989] and Real-Time Optimization systems [Brosilow and Zhao, 1988]. The vector of adjustable parameters, $\boldsymbol{\beta}$ (the model update term), is expected to keep the blending model current such that the model sufficiently represents the blending process with the changing feedstock qualities. The purpose of model updating, as explained briefly in §1.1, is to ensure that the optimization of the updated model gives the true process optimum. It has been shown that the bias updating structure will not necessarily iterate to the true optimum in the presence of disturbances in the constraint parameters (feedstock qualities), [Forbes and Marlin, 1994]. Thus, the LP + bias approach is not an appropriate approach for gasoline blending optimization.

1.3.2 NLP + Bias

The LP + bias form has structural mismatch, because the linear constraints use approximations of the nonlinear blending models. To remove the structural model mismatch, while retaining the simplicity of the bias updating strategy, the Non-Linear Programming with bias updating (NLP + bias) form is sometimes used. Non-Linear Programming is a more general form of mathematical optimization where both the objective function and the constraints can be non-linear functions of the decision variables (Edgar and Himmelblau, 1988; Gill *et al.*, 1981). In this formulation, more accurate non-linear blending models are used (*e.g.*, Ethyl RT-70 for octane blending [Healy *et al.*, 1959]), thus improving the reliability and performance of the blend optimizer. Mathematically, the NLP + bias structure is formulated as:

$$\begin{array}{ll} \max\limits_{\mathbf{x}} & p\mathbf{1}^T\mathbf{x} - \mathbf{c}^T\mathbf{x} \\ \textit{subject to:} & \\ & \mathbf{g}(\mathbf{Q},\mathbf{x}) \le \left[\mathbf{1}^T\mathbf{x}\right]\mathbf{s} - \boldsymbol{\beta} \\ & \mathbf{H}\mathbf{x} \le \boldsymbol{\rho} \end{array} \qquad (1.4)$$

where: **Q** is the quality matrix containing the feedstock qualities, and **g** is the vector of nonlinear functions in the property specification constraints.

In this blend optimization algorithm, the geometry of the true process is more accurately represented than in the LP + bias case. However, the problem of time-varying blend feedstock qualities is not tackled directly by the bias model updating strategy. The algorithm above is based on the current/nominal operating conditions and thus, only optimizes over the current RTO interval. Besides, by treating the feedstock qualities as fixed parameters in the quality constraints, no attempt is made to account for future trends in time-varying parameters. Neither does the NLP + bias approach look at the qualities of the blended pool produced so far while calculating the current feedstock flows.

1.3.3 Time-Horizon Optimizer

As seen above, the bias update model structure does not adequately address the issue of stochastic disturbances in the feedstock qualities. This is due to the fact that the feedstock qualities are treated as fixed parameters in the optimizer while they vary in the actual blending process.

Gasoline is usually blended in batches over a specified length of time. If perfect knowledge of the qualities is given over the length of the blend, an ideal optimizer can be designed [Singh, 1997] that operates the blender in the most profitable manner while ensuring that the blend quality specifications are met at the end of the blend. Even though knowledge of future feed qualities is not possible, past values can be used to predict these future qualities and the predictions can be used to calculate feedstock flowrates required to meet quality specifications. In the Time-Horizon RTO (THRTO) proposed by Singh [1997], future disturbances are predicted over the blend length and future control moves calculated that also consider the properties and amount of gasoline blended up until that point. The feedstock flowrates (control moves) calculated for the current time interval are then implemented. This principle of prediction and optimization to calculate the future moves with only the current feedstock flows (recipe) being actually implemented, is known as receding horizon control and is used extensively in Model Predictive Control [Garcia *et al.*, 1989]. A key difference between MPC and Time-Horizon Real-time Optimization (THRTO) proposed by Singh [1997] is that unlike MPC, only the prediction horizon is user tunable in the THRTO, while the control (optimization) horizon is fixed as the length of time left to the completion of the blend.

The mathematical representation of the Time-Horizon blend optimizer is given as:

$$
\begin{aligned}
\max_{\mathbf{x}} \quad & \sum_{t=t_p}^{t_f} \left(p\mathbf{1}^T\mathbf{x}_t - \mathbf{c}^T\mathbf{x}_t\right) \\
\textit{subject to:} \quad & \\
& \sum_{t=t_o}^{t_p-\Delta t} \mathbf{g}\left(\mathbf{Q}_t,\mathbf{x}_t\right) + \sum_{t=t_p}^{t_f} \mathbf{g}\left(\widehat{\mathbf{Q}}_t,\mathbf{x}_t\right) \leq \left[\sum_{t=t_o}^{t_f} \mathbf{1}^T\mathbf{x}_t\right] \mathbf{s} \\
& \mathbf{g}\left(\widehat{\mathbf{Q}}_t,\mathbf{x}_t\right) \leq \left[\mathbf{1}^T\mathbf{x}_t\right] \mathbf{s} \\
& \mathbf{H}\mathbf{x}_{t_o \Longrightarrow t_f} \leq \boldsymbol{\rho}
\end{aligned}
\tag{1.5}
$$

where: $\mathbf{x}_t$ is the feedstock flowrate vector at time t, $\widehat{\mathbf{Q}}_t$ is the matrix of predicted feedstock properties (uncertain) at time interval t, t_o is the initial blend time (when blending starts), t_p is the present time (when the optimizer runs), Δt is the time interval between consecutive RTO executions and t_f is the final blend time (blending done). In this formulation, it is assumed that the feed properties remain constant over every RTO interval (*e.g.*, t_p to t_{p+1}).

The objective function to be maximized in Problem 1.5 is the total profit over the blend horizon. The first constraint represents the satisfaction of the quality constraints at the end of the blend, while the second constraint states that the current recipe must also be within specifications. In the event of blending being halted before the scheduled end of blend, the second constraint reduces the possibility of the gasoline blended so far being off-specification. The final constraint is the condition that the demand and availability constraints are satisfied over the blend.

1.3.4 Commercial Packages

There are a number of software packages available commercially for gasoline blending scheduling and control [Anonymous, 1997]. All of these packages attempt to integrate blend scheduling and blend optimization. They do this by sequencing the blends and then trying to enforce the scheduled recipes while minimizing giveaway. Many of these packages claim to use non-linear programming (NLP) techniques in the Blend Optimizer. However, for most of them the non-linear part is probably a quadratic term in the objective function penalizing deviation from scheduled recipe, while the property constraints are linear. As shown in Singh [1997], using linear blending models can lead to major losses in performance for the on-line optimizer. Also, in none of the software packages is uncertainty in any of the parameters in the blending optimization problem taken into account. There are some vendors who use more accurate non-linear blending models and Near Infra-Red (NIR) analyzers [Anonymous, 1997] to update the blended and/or feedstock qualities in the model, but none of them takes the uncertainty in or reliability of the measurements into account.

1.3.5 Discussion of Available Methods

As seen in sections 1.3.1 and 1.3.2, problems such as plant/model structural mismatch (for LP + bias) and disturbances in feedstock qualities (both LP + bias and NLP + bias assume constant feedstock qualities) are not handled by the gasoline blending optimization approaches based on bias updating.

The constant feed qualities assumption is valid only if the blending is done out of well mixed, standing feedstock tanks. These days many refineries are using running tanks *i.e.*, fresh feedstock is being pumped into the tank from the upstream process even as feedstock is being withdrawn from the tank for blending. This is done in an effort to reduce costs by minimizing inventories. As feedstock qualities are dependent on the process conditions in the unit they are coming from, the qualities/properties of these streams change with changing upstream process operations. Thus, feedstock qualities are time-varying and stochastic in character and, change continuously with upstream operation changes. To complicate matters further, all quality measurements (for feedstocks as well as final blends) are subject to measurement noise. Using the LP (or NLP) formulations with bias updating, while assuming the qualities to be constant at given expected values, can give setpoints for the feedstock flow-rates that may violate one or more of the specification constraints. Constraint violation in actual industrial blenders is usually avoided by using constraint controllers below the blending RTO system. The objective of these controllers is to minimize quality giveaway and deviation from blend recipes (passed down from the RTO) while trying to ensure that the quality constraints are not violated. Thus, when the blender starts to produce off-specification blends, the constraint controller deviates from the "optimal" recipe to make the blend on-specification. The recipe implemented might not deviate from the "optimal" recipe in the most economic manner and makes the final blend less profitable.

The costs of most of the gasoline feedstocks are not known with certainty. This is true especially of those feedstocks that are not sold as such and are refinery intermediate products (*e.g.*, FCCU Gasoline) which are used as feedstocks for final products that are sold. These feedstock costs are decided based on some heuristics by the refinery scheduling department. In some situations the demand for the blends and/or availability of the feedstocks may change while the blending is under way. This translates to uncertainty in these parameters. Therefore, a key problem refineries face is formulating the blending optimization and control problem such that the uncertainty in the all the above mentioned parameters has the least possible effect on the optimality of the solution, and does not cause the optimizer to generate infeasible solutions.

1.4 Thesis Objectives and Scope

This thesis is concerned with the design of the RTO layer in a Gasoline Blending Control system. As discussed in the previous sections, the RTO layer in current blend controllers is unable to handle stochastic disturbances in the feedstock qualities satisfactorily. To minimize blending of off-specification gasoline, most industrial blenders accept some product quality give-away [Grosdidier, 1997]. Current practise is to blend to targets which exceed specifications for the more critical qualities (*e.g.*, octane, RVP, *etc.*) by some arbitrarily decided amount [Treiber *et al.*, 1998]. The main objective of this thesis is to design an RTO based blend optimizer that directly incorporates parametric uncertainty, and can effectively handle stochastic disturbances in blend feedstock qualities. Such a blend optimizer would thus be a systematic method to calculate the most economically effective manner to exceed specifications or provide an acceptable probability of feasibility.

Stochastic Programming is the branch of Mathematical Programming concerned with optimization problems that have uncertain parameters. A brief introduction to Stochastic Programming is given at the beginning of Chapter 2. In the Gasoline Blending Optimization Problem, all model parameters: economic, property data, demand/availability, are uncertain to some degree. The scope of this thesis is limited to developing strategies to deal with uncertainty in gasoline feedstock qualities (constraint parameters in blending optimization) only. To this end, various Stochastic Programming methods that deal with uncertainty in constraint parameters in optimization problems are reviewed. Solution techniques for solving these problem formulations that incorporate uncertainty information directly are discussed. The suitability of these different techniques for implementation in a Real-time Optimization system is then examined. Based on the above-mentioned discussions, the Stochastic Programming formulations used in this thesis for Gasoline Blending Optimization are decided upon.

The discussion on the Gasoline Blending Optimization problem is continued in Chapter 3. Blending models actually used in this work are described. Uncertainty issues in gasoline blending, especially uncertainty in feedstock qualities, are examined in more detail. This is followed by development of algorithms for gasoline blending optimization that directly incorporate parametric uncertainty in the blending model constraints. Chapter 4 develops the case study used to demonstrate the effectiveness of the proposed algorithms in dealing with stochastic disturbances in gasoline feedstock qualities. The results of the case study are then presented and the performance of the different optimizers is compared. Chapter 5 makes some conclusions as to the effectiveness of the strategies employed to deal with parametric uncertainty in gasoline blending optimization. Some implications for the wider area of Real-Time Optimization are discussed and possible future directions of research are considered.

The Stochastic Programming formulations used in this thesis are applicable to any optimization problems, on-line or off-line, where some optimization parameters are not known with complete certainty.

Chapter 2

Stochastic Programming for Uncertain Constraint Parameters

"Stochastic Programming handles mathematical programming problems where some of the parameters are random variables ..." [Prekopa, 1995] is one of the simpler definitions given for Stochastic Programming (SP). Stochastic Programming or Stochastic Optimization came into being in the mid 1950s' [Dantzig, 1955]. The field was born when people first started to think about what to do in case some of the parameters in a linear (or non-linear) optimization problem were uncertain. Thus, Stochastic Programming consists of formulating optimization problems that directly incorporate uncertainty in the optimization parameters and investigating techniques to solve such problems.

2.1 Introduction

Consider the general optimization problem (Problem 2.1 below).

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & f(\mathbf{c},\mathbf{x}) \\ \textit{subject to:} & \\ & \mathbf{h}(\mathbf{x},\boldsymbol{\xi}) \geq \mathbf{b} \end{array} \tag{2.1}$$

where: f is the objective function, $\mathbf{c}$ is the vector containing economic data, $\mathbf{x}$ is the decision variables vector, $\mathbf{h}$ is the vector of inequality constraints, $\boldsymbol{\xi}$ is the set of constraint parameters, and $\mathbf{b}$ is the right hand side vector of the inequality constraints. Equality constraints are not considered in the above formulation of Problem 2.1 as it has been assumed that any equality constraints have already been used to reduce the dimensionality of the optimization problem (*i.e.*, the optimization problem is considered in its reduced space).

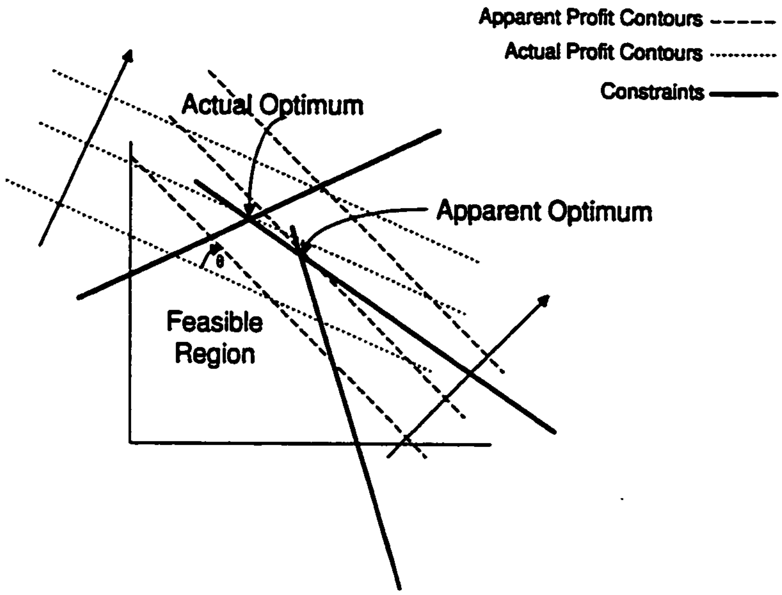

Figure 2.1: Illustration of uncertainty in objective function (cost data)

Stochastic Programming problems can be split into three categories based on the location of uncertainty in the optimization problem:

1. uncertainty in economics or the objective function parameters (**c**).

2. uncertainty in the right hand side elements of the constraints (**b**).

3. uncertainty in the constraint function parameters ($\boldsymbol{\xi}$).

To aid visualization, the above three SP categories are illustrated by showing the effect of uncertainty on an LP problem in two dimensions. All the arguments presented below can be generalized to an n-dimensional non-linear programming (NLP) problem.

Uncertainty in economic data translates to uncertainty in the gradient (slope and direction) of the profit surface. As seen in Figure 2.1, this uncertainty in the slope of the objective (profit) function can cause the apparent optimum[1] to shift to a sub-optimal operating point[2]. If the angle of tilt of the profit surface changes by θ (from the nominal value), the calculated optimum (apparent optimum) shifts to a new 'sub-optimal' point.

[1] Apparent optimum: the solution calculated using nominal values (vs actual values) for the uncertain parameters.

[2] sub-optimal with respect to the actual plant or system

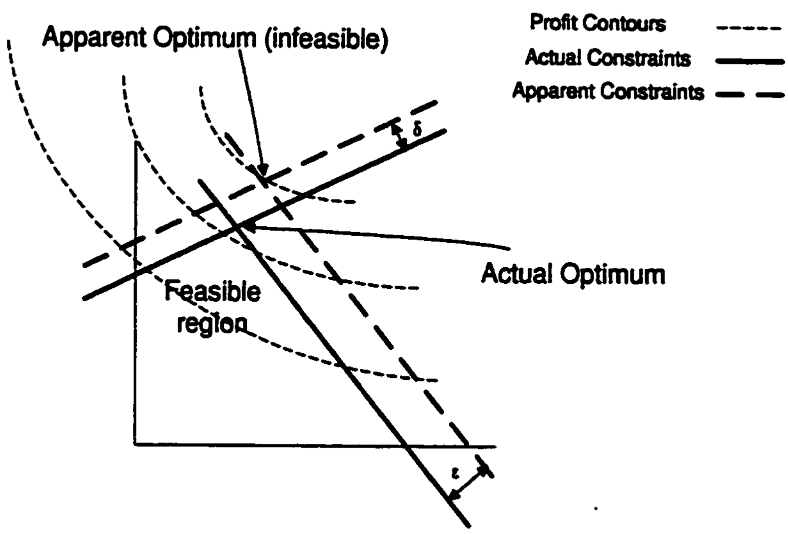

Figure 2.2: Illustration of uncertainty in constraint parameters I: Uncertain RHS

The second category of Stochastic Programming problems is that of optimization problems with uncertain parameters in the right hand vector (**b**) of inequality constraints. Geometrically, uncertainty in the **b** vector leads to shifting or lateral translation of the inequality constraints (Figure 2.2). An example of a physical parameter in the right hand side vector for the inequality constraints is the availability of a feed-stream in a process. Thus, changes in feedstock availability can shrink or expand the region of feasible solutions. This uncertainty in feasible region size implies that solutions generated using the nominal values for the uncertain b_i could lie outside the actual feasible region. As illustrated in Figure 2.2, lateral shifting (by δ and ε) of the active constraints causes the apparent optimum to move into the infeasible region. Thus, in this case, ignoring the uncertainty in the **b** vector would cause the optimizer solution to be infeasible with respect to the actual physical process. Shifts in the opposite direction to that shown in Figure 2.2 will result in the apparent optimum moving into the feasible region. This would lead to a feasible, but sub-optimal, solution.

In the third category of Stochastic Programming problems, lie problems where the constraint function parameters are uncertain ($\boldsymbol{\xi}$ in Problem 2.1). This uncertainty translates to uncertainty in both or either of the slope and the shape (curvature) of the constraints. For a linear model, only the slope of the (linear) constraints would be uncertain. This results in uncertainty in the shape of the whole feasible region (governed by uncertain ϕ and γ in Figure 2.3) defined by the quality constraints. This uncertainty in the shape and size of the feasible region again implies that op-

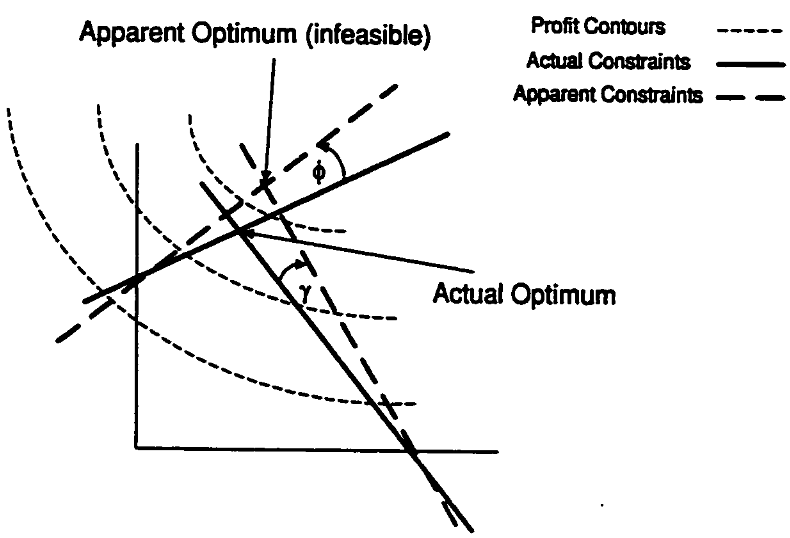

Figure 2.3: Illustration of uncertainty in constraint parameters II: Uncertain LHS coefficients

timizer solutions generated using the nominal (expected) values for the parameters ξ_i, can either lie outside the actual feasible region[3], or be sub-optimal by moving into the feasible space, depending on the direction of deviation in ξ_i. Examples of constraint function parameters for the gasoline blending optimization problem are all the feedstock qualities/properties.

The work in this thesis is restricted to problems that fall into the third category (*i.e.*, uncertain constraint coefficients), because as stated above, feedstock qualities are constraint function parameters in the gasoline blending optimization problem. In this chapter, Stochastic Programming formulations for treating uncertainty/disturbances in constraint parameters (both b and ξ) are introduced and discussed.

2.2 Different formulations for Uncertain Constraints

Uncertain constraint parameters in an optimization problem can cause infeasible solutions to be generated when the problem is solved using expected values of the uncertain parameters (Figures 2.2 and 2.3). Implementation of infeasible solutions can cause a number of process problems. Examples of such problems include safety and environmental code violations, damage to processing equipment, production batches

[3]actual feasible region: the subspace defined by actual physical constraints where there is no parameteric mismatch in ξ.

that cannot be sold on the market (either discarded or need corrective measures before they can be sold), and so forth.

Optimization problems with uncertain parameters in the constraints can be dealt with in a number of ways. Discussions in this chapter will assume all inequality constraints have stochastic parameters. If a specific problem also contains strictly deterministic constraints, these deterministic constraints would remain as deterministic constraints in the Stochastic Programming formulation. Some of the possible formulations developed to deal with uncertain constraint parameters are discussed below. Each formulation has a different aim or objective, but all of them attempt to minimize the effect, in some sense, of uncertainty in constraint parameters on the "optimal" solutions generated.

2.2.1 Probability Maximization

In the probability maximization formulation [Prekopa, 1995], there is no explicit economic objective function. This form is used in situations where a "safe" operating region is required to ensure that the risk of any constraint violations is low. The objective function to be maximized is the probability of the uncertain constraints being satisfied:

$$\max(Prob(\mathbf{h}(\mathbf{x},\boldsymbol{\xi}) \geq \mathbf{b})) \tag{2.2}$$

where *Prob* stands for probability.

This formulation requires the solution to be such that the probability of the constraints being violated is the least out of all possible solutions. As seen in Problem 2.2, economics are not used explicitly, and the sole aim is to minimize the probability of infeasibility. Implicitly though, an economically optimal solution is generated as violation of the uncertain constraints has a cost associated with it. This formulation would be appropriate in situations where the violation of constraints has a very high cost associated with it and the "safest" operating point/region in the presence of uncertain constraint parameters is required. However, there may be many operating points with a high safety margin. Thus, a better objective would be to search for a solution that takes the economics into account explicitly, while requiring a high margin of safety.

2.2.2 Penalty Function forms

Another method of posing the problem for uncertain constraints is the penalty function method (Problem 2.3). For this formulation, the uncertain constraints are incorporated into the objective function by penalizing their infeasibility. A function q is defined which measures the violation of the stochastic constraints and is included in the objective function with the economic objective to be minimized. One of the main problems with this approach is that the quantification of cost of infeasibility is not easy. Another is that the calculation of the expectation of the penalty function is non-trivial.

$$\min_{\mathbf{x}} f(\mathbf{c}, \mathbf{x}) + E\left[q(\mathbf{b} - \mathbf{h}(\mathbf{x}, \boldsymbol{\xi}))\right] \tag{2.3}$$

where: q is a penalty function on constraint violation

As an example, if the penalty function is simply the magnitude of the violation, the blending optimization problem with uncertain feedstock $RVPs$ can be written as:

$$\min_{\mathbf{x}} f(\mathbf{c}, \mathbf{x}) + E\left[(RVP_{blend} - RVP_{low\_spec}) \mid (RVP_{blend} \geq RVP_{low\_spec})\right] \tag{2.4}$$

The above form minimizes the cost of producing the blend and the expected violation of the uncertain constraint. Since penalty function approaches cannot ensure feasible solutions for finite values of the penalty function, barrier functions [Fletcher, 1987; Gill *et al.*, 1981], which would ensure feasibility may be used in this formulation.

2.2.3 Programming under Probabilistic Constraints

Instead of directly minimizing the probability of an infeasible solution (§ 2.2.1), another approach could be to set a reliability level for the uncertain constraints by requiring that they be satisfied at a minimum (high) level of probability. In such a formulation, the original economic objective function can be retained as such, while the uncertain constraints are now framed as probabilistic constraints [Charnes and Cooper, 1963], [Vajda, 1972]. A solution to this Probabilistic Programming problem would minimize economic cost while ensuring that the probability of the constraints being violated is very small (the allowed probability of violation is specified during problem formulation). Thus, the strategy adopted is to optimize the economic advantage, while keeping the risk of constraint violation due to uncertainty at acceptably low levels.

Probabilistic Programming can be further divided into two categories: 1) programming under individual probabilistic constraints and 2) programming under joint probabilistic constraints.

Individual Probabilistic Constrained Programming

In this formulation [Charnes and Cooper, 1963], each individual constraint (with uncertain parameters) is converted into a probabilistic constraint with an associated probability level of feasibility. The risk minimization takes the form of ensuring that each uncertain constraint is satisfied with a probability of at least π, where $\pi \in (0,1)$. The individually constrained, probabilistic programming problem is:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & f(\mathbf{c},\mathbf{x}) \\ \textit{subject to:} & \\ & Prob\{h_i(\mathbf{x},\boldsymbol{\xi}) \geq b_i\} \geq \pi_i \text{ for } i = 1,...,n \end{array} \tag{2.5}$$

where: h_i is the i^{th} inequality constraint function, and b_i is the right hand side of the i^{th} inequality constraint.

As an example of the formulation of individual chance constraints, consider the maximum Reid Vapor Pressure constraint on blended Premium Gasoline. If the Reid Vapor Pressures of the feedstocks are uncertain, and the specification on $RVP_{premium}$ states it cannot be more than 10.8 *psi*, the $RVP_{premium}$ chance constraint would be formulated as:

$$Prob(RVP_{premium} \leq 10.8) \geq 0.95 \tag{2.6}$$

where the minimum probability of feasibility is taken as 95% for this example.

This is the formulation used in the remaining part of this work. Constraints of this type, where each constraint has to be satisfied at least $\pi \times 100\%$ of the time, are called individual chance constraints [Charnes and Cooper, 1963], and the formulation Chance Constrained Programming or CCP.

Joint Probabilistic Constraints

Only individual probabilistic constraints (CCP) were considered above. Such formulations ensure that each constraint is violated no more than $(1-\pi) \times 100\%$ of the time. However, in actual practise it may be reasonable to specify that none of the uncertain constraints are violated $(1-\pi) \times 100\%$ (a small percentage) of the time. Consider a set of constraints with uncertain parameters. If violation in any of these is unacceptable, it would be desirable to frame the problem as one in which the probability of the whole set of constraints being satisfied is always greater than some minimum risk threshold.

Probabilistic constraints of this type are known as Joint Probabilistic Constraints [Miller and Wagner, 1965]. The new optimization formulation with Joint Probabilistic Constraints (JPC) can be written as:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & f(\mathbf{c},\mathbf{x}) \\ \textit{subject to:} & \\ & Prob\{\mathbf{h}(\mathbf{x},\boldsymbol{\xi}) \geq \mathbf{b}\} \geq \pi \end{array} \tag{2.7}$$

For a simple illustration of the idea, consider the set of quality constraints in the general blending optimization problem (Problem 1.1). Instead of formulating the *RON*, *MON*, and *RVP* constraints separately as individual chance constraints, they are lumped into one probabilistic constraint which states that the whole set of constraints with uncertain parameters must be satisfied at a probability level of π. This would be formulated as:

$$Prob \begin{bmatrix} RON_{blend} \geq RON_{low\_spec} \\ MON_{blend} \geq MON_{low\_spec} \\ RVP_{blend} \leq RVP_{high\_spec} \end{bmatrix} \geq 0.95 \tag{2.8}$$

As seen above, the JPC formulation is a more natural approach for handling the uncertain constraint parameters problem. A joint probability criterion can be met by using individual probabilistic constraints; however, the tuning of the probability levels for the individual chance constraints to meet this JPC criterion is non-trivial (refer to Appendix C for a more complete discussion).

Probabilistic constraints are normally used where the cost of infeasibility is very high (and hard to ascertain), thus making the use of penalty function approaches (§ 2.2.2) difficult. The assigning of values for π, the minimum probability of feasibility, is heuristic except for some situations where the acceptable risk threshold is defined in the specifications for the product. An example of such a situation could be the manufacture of simple high volume hardware products such as nuts and bolts. There are specifications on the physical dimensions of these products. In most cases there is also a small, specified, acceptable fraction of production volume which may be defective. This fraction of acceptable defects can be used while selecting the operation sequence and the operating points of the machinery used to manufacture these products.

Although Probabilistic Programming limits the probability (or frequency) of infeasible solutions, the degree or extent by which the uncertain constraints are violated is not considered. This requirement of somehow limiting the degree of infeasibility for the solutions that are infeasible can be met using different techniques, such as the one outlined in the next section.

2.2.4 Constraints with Conditional Expectations

A formulation that can deal effectively with optimization problems where the extent of violation of uncertain constraints is important, is the Conditional Expectations form [Prekopa, 1973]. In the formulation of this problem, the constraints state that if the i^{th} uncertain constraint is violated, the expected violation is to be bounded in magnitude by a specified positive number d_i. The mathematical formulation of the conditional constraints problem is given as:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & f(\mathbf{c},\mathbf{x}) \\ \textit{subject to:} & \\ & E[\mathbf{b}-\mathbf{h}(\mathbf{x},\boldsymbol{\xi})|\mathbf{h}(\mathbf{x},\boldsymbol{\xi})<\mathbf{b}] \leq \mathbf{d} \end{array} \tag{2.9}$$

where: **d** is the set of numbers (vector) that gives the upper bounds on the expected constraint violations. **d** can also be called the vector of maximum expected violations.

In this form, there is no bound on the probability or frequency of constraint violation, but there is an explicit bound on the extent of violation. Even though the solution to the conditional constraints problem may violate constraints at a high probability level (or a high frequency), the amount by which the constraints are violated is small (specified). For example, consider again the maximum Reid Vapor Pressure constraint for Premium Gasoline in the blend optimization problem (Problem 1.1). As in the previous subsection (§ 2.2.3), the premium gasoline *RVP* specification bounds $RVP_{premium}$ to less than or equal to 10.8 *psi*. If a final blended *RVP* of 10.82 *psi* is considered to be within acceptable limits, a conditional constraint could be formulated for this case as follows:

$$E[(RVP_{blended,premium} - 10.8) \mid (RVP_{blended,premium} > 10.8)] \leq 0.02 \tag{2.10}$$

The above inequality states that if the *RVP* of the premium grade being blended falls above 10.8 *psi*, the expected value of this off-specification blend should be below 10.82 *psi*. Thus for this case, the maximum allowed average deviation from the $RVP_{premium}$ specification is 0.02 *psi*.

Constraints of this type would be useful in problems where the "distance from feasibility" or maximum extent of infeasibility of the solution is very important (*e.g.*, situations where "small" constraint violations are acceptable).

2.2.5 Combinations of the above forms

The last three techniques for treating uncertain constraint parameter problems (§ 2.2.2 → § 2.2.4) can also be combined to give formulations which specify multiple objectives. Some of these combinations are given below and explained briefly.

A Probabilistic Programming problem with Conditional Expectation Constraints can be represented as:

$$\begin{aligned} &\min_{\mathbf{x}} && f(\mathbf{c},\mathbf{x}) \\ &\textit{subject to:} && \\ & && Prob\{\mathbf{h}(\mathbf{x},\boldsymbol{\xi}) \geq \mathbf{b}\} \geq \pi \\ & && E[\mathbf{b} - \mathbf{h}(\mathbf{x},\boldsymbol{\xi}) | \mathbf{h}(\mathbf{x},\boldsymbol{\xi}) < \mathbf{b}] \leq \mathbf{d} \end{aligned} \qquad (2.11)$$

In Problem 2.11, joint probabilistic constraints and constraints with conditional expectations are applied to the same set of uncertain constraints simultaneously. The first constraint in Problem 2.11 would ensure that the original uncertain constraints (Problem 2.1) are satisfied at a probability level of at least π. The second constraint in Problem 2.11 above states that if those original uncertain constraints are in fact violated, the average violations are limited in magnitude by the vector $\mathbf{d}$. Thus, both the frequency and the magnitude of constraint violations in the presence of parametric uncertainty can be limited by a combined Probabilistic Programming and Conditional Expectation Constraints formulation.

Different Stochastic Programming techniques that handle parametric uncertainty in optimization problems have been discussed in this chapter. The solution to each of these problems requires knowledge of the probability distribution of the uncertain parameters. Some of the more commonly seen characterizations for parametric uncertainty are continuous probability distributions *e.g.*, Normal or Gaussian and uniform or flat probability distributions. Other possibilities include discrete distributions where the uncertain parameters are only allowed to take values in fixed, finite sets.

2.2.6 Permanently Feasible Linear Programs

In this approach [Friedman and Reklaitis, 1975], a solution strategy is derived for LPs, where some or all of the constraint parameters are uncertain. It is assumed that the disturbances in the constraint parameters are uniformly and independently distributed over some known range of variation $\pm\alpha$. This method was proposed for processes where no information is available about the nature of the statistical distribution of the constraint parameters in a linear program. The only description assumed available is mean values of the parameters and the expected range of variation in each parameter. In such a scenario the assumption of a uniform distribution for the uncertain parameters may be appropriate. In this method, a worst case scenario design of the LP is done, where worst case values (the values which would drive the nominal solution furthest into infeasibility) are assigned to all uncertain parameters. This ensures that the modified LP solution is feasible for the worst case variations in the uncertain parameters. The solution thus obtained is feasible over all possible variations (within the range given) in the uncertain parameters. As an example, assume that the Reid Vapour Pressure of the Reformate stream varies by as much as ± 0.3 *psi* (from a nominal value of 3.8 *psi*). In this situation, the LP to be solved for feedstock flowrates would use a value of 4.1 *psi* for the Reformate RVP in the maximum RVP constraint. This would ensure that the solution (feedstock flows) will not violate the maximum RVP constraint for all values of reformate RVP ≤ 4.1 *psi*. The LP so designed can thus be called the Permanently Feasible LP[4] (PFLP).

If there is some known dependence between variations in the parameters (*e.g.*, if the RON and MON of any or all feedstocks vary together), this dependence can be incorporated into the problem form and solved to yield a less conservative solution (sometimes) than that for the independent variations case.

Consider the following LP:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{c}^T\mathbf{x} \\ \textit{subject to:} & \\ & \mathbf{A}\mathbf{x} \leq \mathbf{b} \end{array} \tag{2.12}$$

If α is the matrix of the maximum allowed variations *i.e.*, $a_{i,j}^{true} = a_{i,j} \pm \alpha_{i,j}$, where all $\alpha_{i,j}$ are positive and independent of each other, the PFLP formulation for the above problem is given by:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{c}^T\mathbf{x} \\ \textit{subject to:} & \\ & (\mathbf{A} + \alpha)\mathbf{x} \leq \mathbf{b} \end{array} \tag{2.13}$$

[4]Permanently Feasible Linear Program is a non-standard term used here to represent LPs' formulated as given in §2.2.6. These LPs are refered to as Flexible Programs in Friedman and Reklaitis [1975].

If the disturbances in the parameters of the constraint matrix are known to be column dependent (*i.e.*, disturbances in some of the parameters in a column of the constraint matrix are linearly dependent), there is a possibility of making the PFLP solution less conservative. In this case, a vector $\boldsymbol{\mu}_j$ can be defined for each column of the $\boldsymbol{\alpha}$ matrix which represents the dependencies of the elements of that column as:

$$\boldsymbol{\mu}_j^T \boldsymbol{\alpha}_j = 0 \tag{2.14}$$

where j is the column (for our purposes the feedstock) index, and $\mu_j \neq 0$ for all j.

For the column dependent form, the modified LP is given by [Friedman and Reklaitis, 1975]:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{c}^T\mathbf{x} \\ \textit{subject to:} & \\ & (\mathbf{A}+\boldsymbol{\chi})\mathbf{x} \leq \mathbf{b} \end{array} \tag{2.15}$$

where $\chi_{i,j} = \left\{ \begin{array}{l} \min(\alpha_{i,j}, \sum\limits_{\substack{k=1 \\ k \neq i}}^{M} \left| \frac{\mu_{k,j}\alpha_{k,j}}{\mu_{i,j}} \right|, \text{ if } \mu_{i,j} \neq 0 \\ \alpha_{i,j}, \text{ if } \mu_{i,j} = 0 \end{array} \right\}$, where M is the number of constraints containing uncertain parameters, and $\alpha_{i,j}$ is the maximum perturbation allowed in the j^{th} element of the i^{th} constraint. $\mu_{i,j}$ is given by equation (2.14).

If the above approach (PFLP) is applied to a problem where the parameter variations are assumed to be independent and normally distributed, $\boldsymbol{\alpha}$ can be specified as some multiple, n, of the known or estimated standard deviations of the parameters: $\boldsymbol{\alpha} = n\boldsymbol{\sigma}_{disturbance}$. It should be noted that this does not imply that the resulting constraints now have a probability of feasibility given by $F^{-1}(n)$, where F is the normal probability distribution function (although the above would be true in the case of there being only one uncertain parameter in each constraint).

2.3 Discussion

Some important Stochastic Programming techniques to handle uncertain constraint parameters have been described in the section above (§2.2). Solution methods to solve these problem forms are discussed below and their suitability is examined for application to RTO systems in general and Gasoline Blending Optimization in particular. The discussions of solution techniques are only valid for problems where the original problem is an LP:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{c}^T\mathbf{x} \\ \textit{subject to:} & \\ & \mathbf{A}\mathbf{x} \geq \mathbf{b} \end{array} \tag{2.16}$$

where: $\mathbf{A}$ and $\mathbf{b}$ can have stochastic elements.

2.3.1 Solution Methods

To solve the Probability Maximization and Joint Probabilistic Programming problems (Problems 2.2 and 2.7), a variety of non-linear optimization algorithms can be used [Prekopa, 1995] (*e.g.*, generalized reduced gradient (GRG) methods [Gill *et al.*, 1981], Interior Point methods utilizing logarithmic barrier functions [Wright, 1997] , *etc.*). However, the calculation of numerical values for both the objective function and its gradient require application of multivariate numerical integration methods (discussed in some detail in Prekopa [1995]). This in turn means that the solution techniques for Probability Maximization and Joint Probabilistic Programming have to incorporate a large number of simulation runs, multi-dimensional numerical integration, and non-linear programming making the solution to these problems computationally very expensive. The usual approach [Terwiesch *et al.*, 1998] is to discretize the probability density functions of the uncertain parameters and numerically integrate to get an approximate solution to the original problem where the distribution functions were continuous (*e.g.*, multivariate normal distribution).

The special case of the Probabilistic Programming problem with individual chance constraints for a Linear Program can be expressed as:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{c}^T\mathbf{x} \\ \textit{subject to:} & \\ & Prob(\mathbf{a}_i\mathbf{x} \leq b_i) \geq \pi_i \text{ for } i = 1, ..., n \end{array} \tag{2.17}$$

where: $\mathbf{c}$ is the vector containing economic data, $\mathbf{x}$ is the decision variables vector, $\mathbf{a}_i$ is the coefficient vector for the i^{th} inequality constraint, and b_i is the right hand side of the i^{th} inequality constraint.

Chance Constrained Programming (individual chance constraints) has been used to obtain solutions robust to uncertainty in linear program constraint parameters in areas as diverse as Optimal Polymer Design and Chicken-feed Formulation. The Polymer Design problem consists of using Computer-Aided Product Design techniques [Constantinou *et al.*, 1996] to identify possible molecular structures that will have some specified desired properties. Group contribution methods [van Krevelen, 1990] are used to predict properties of different molecular structures. However, as there is always uncertainty in these predictions, an optimal polymer design problem using these methods needs to incorporate this uncertainty into the optimal design problem [Maranas, 1997]. Similarly, in Animal Feed Formulation, the nutrient concentrations in various types of animal foods can vary. Therefore, the formulation of an optimal animal feed recipe from these feedstocks takes the form of an optimization problem (LP) where the constraints parameters are uncertain. The animal feed formulation has also been solved using CCP and the benefits are outlined in Roush *et al.* [1994]. More recently, Chance Constrained Programming has also been applied in the area of Model Predictive Control (MPC) [Schwarm and Nikolaou, 1999]. A typical MPC formulation is an optimization problem that uses predicted process outputs over a time horizon. These predictions are parameters in the linear inequality constraints of the optimization problem formed. Thus, as illustrated in [Schwarm and Nikolaou, 1999], the incorporation of probabilistic process output constraints in an MPC formulation leads to solutions that are much more robust with respect to the uncertainty in the process model.

Problem 2.17 (with the assumption $\mathbf{a}_i$, b_i have a multivariate normal distribution) can be converted to a deterministic, quadratically constrained problem [Vajda, 1972] as shown in §3.3.2 for the blending optimization problem. This deterministic problem can be solved using any non-linear optimization routine. Recently, Boyd and coworkers have used Second-order Cone Programming (SOCP) to solve CCP problems. Second-order Cone Programming [Lobo *et al.*, 1997] uses interior point methods [Wright, 1997] to solve what they call Robust Linear Programming (LPs with uncertainty in constraint parameters) problems.

An approximate solution to the Joint Probabilistic Programming problem can be obtained by combining Individual Chance Constrained Programming with some probability bounding techniques. Probability bounding and approximation techniques based on Boole's inequality, which can be used to get bounds for the joint probability of feasibility level, are given in Appendix C. However, the solutions obtained using this combination cannot be guaranteed to be optimal because the joint probabilities problem is addressed in an indirect manner. Thus, optimality is compromised in return for low risk solutions. Also, the bounded solutions to the JPC problem thus obtained are non-unique, as there are an infinite number of individual probability levels that can be specified to get the same bounds on the joint probability level of feasibility.

The Conditional Constraints and Penalty Function forms require calculation of mathematical expectations of functions with random parameters. This again involves integration over multivariate probability density functions, which can be highly computationally intensive.

The last method presented in §2.2.5, Permanently Feasible Linear Programs, can be solved using regular linear programming techniques as the formulation is that of a standard LP.

2.3.2 RTO Implementation Issues

In closed loop Real-Time Optimization systems, optimizer solution times are a very important issue because of the direct repercussions on RTO execution frequency. Solving a problem with a large number of variables with a stochastic programming formulation can be computationally very expensive. Of the SP formulations discussed in this chapter, only the Individual Chance Constrained Programming and the Permanently Feasible LP algorithms are relatively cheap computationally.

For the Gasoline Blending RTO, computational times are a lesser issue because the problem size is small (typically 10-30 optimization variables[5]). Even so, Individual Chance Constrained Programming is particularly suited to this problem as: 1) assessing the cost of producing an infeasible batch is non-trivial thus making the penalty function approach even more difficult, and 2) even though some leeway is allowed in the quality constraints (RON, MON, RVP, etc.), specifying conditional expectation constraints does not seem to justify the additional computation involved.

The PFLP algorithm [Friedman and Reklaitis, 1975] was designed for constraint parameter uncertainty defined by uniform distributions. Although as mentioned in §2.2.5, the algorithm can be applied (by using two σ or three σ limits) for disturbances with normal distributions, this gross approximation of the underlying probability distribution distorts the problem. For the above reasons, only the Individual Chance Constrained Programming is used in this work to design gasoline blending optimizers which incorporate parametric uncertainty explicitly.

[5]Although formulation of the discretized THRTO form (§1.3.3) will increase the number of optimization variables to *(number of feedstocks)*×*(number of RTO intervals to the end of the blend)*.

Chapter 3

Gasoline Blending Optimization

In this chapter, strategies to incorporate uncertainty into the formulations of Real-Time Optimization for gasoline blending are developed. The chapter begins with a brief description of the sets of blending models, both linear and non-linear, used in this work. This is followed by an account of the problems caused by uncertainty/disturbances in blending model parameters. Next, the Stochastic Programming techniques used in this work to counter some of the effects of these uncertainties are elucidated upon. Finally, the extensions of the above techniques to the case of gasoline blending RTO under uncertainty are described.

3.1 Blending Models

There are many properties that are key in characterizing gasoline (*e.g.*, octane number, *RVP*, ASTM distillation points, olefins and aromatics content, flash point, viscosity, *etc.* [Gary and Handwerk, 1994]). Some of the most important qualities, and the only ones considered in this work are Research Octane Number (*RON*), Motor Octane Number (*MON*) and Reid Vapour Pressure (*RVP*). In this work, blending models are used for two purposes; 1) simulation of the blending process (the plant), and 2) framing the constraints (the model) of the RTO based blend optimizer.

The main requirements for the blending models to be used in the optimizer are accuracy, simplicity and parsimony of (model) structure. The models used for the case studies in this work are described below. These models were originally used in Singh [1997] and a more complete discussion of their characteristics can be found there. A review of the different gasoline blending models (for *RON*, *MON* and *RVP*) available in literature can also be found there.

3.1.1 Linear Model

The linear models used here are the blending octane number method for octane numbers and the blending index method for *RVP* blending.

In the blending octane number method, fictitious blending octane numbers (*BON*s') are used instead of the true *RON* and *MON* of the feedstocks. These *BON*s blend linearly (on volumetric basis) to give the octane number of the blend [Gary and Handwerk, 1994]:

$$ON_{blend} = \sum_{i=1}^{n_{comp}} \omega_i \times BON_i \tag{3.1}$$

where: ω_i is the volume fraction of component i, ON_{blend} is the octane number (*RON* or *MON*) of the blend, and n_{comp} is the number of components in the blend.

In the blending index method for *RVP* blending, blended *RVP*s are predicted by using Reid Vapour Pressure Blending Indices (*RVPBI*). *RVPBI*s blend linearly and are calculated as [Gary and Handwerk, 1994]:

$$\begin{aligned} RVPBI_i &= (RVP_i)^{1.25} \\ RVPBI_{blend} &= \sum_{i=1}^{n} \omega_i \times RVPBI_i \end{aligned} \tag{3.2}$$

3.1.2 Non-Linear Model

For the non-linear model, the Ethyl RT-70 method for blending octane numbers is used along with the *RVP* blending indices method for *RVP* blending.

The Ethyl method is one of the more accurate and parsimonious models used to calculate blended octane numbers [Singh, 1997]. In this method, the blended octane numbers are explicit functions of the octane numbers (*RON*, *MON*), component sensitivity (*RON*−*MON*), olefin content, and the aromatic content of the feedstocks [Healy *et al.*, 1959]:

$$\begin{aligned} RON_{blend} &= \bar{r} + a_1(\overline{rs} - \bar{r}\bar{s}) + a_2(\overline{O^2} - \overline{O}^2) + a_3(\overline{A^2} - \overline{A}^2) \\ MON_{blend} &= \bar{m} + a_4(\overline{ms} - \bar{m}\bar{s}) + a_5(\overline{O^2} - \overline{O}^2) + a_6\left(\frac{\overline{A^2} - \overline{A}^2}{100}\right)^2 \end{aligned} \tag{3.3}$$

where: r is *RON*, m is *MON*, s is sensitivity (*RON*−*MON*), O is olefin content (volume %), A is aromatic content (% by volume), and a_1, a_2, a_3, a_4, a_5, a_6 are correlation coefficients. All quantities with over-bars represent volumetric averages.

3.2 Uncertainty in Gasoline Blending

Almost all model parameters in the Gasoline Blending RTO problem:

$$\begin{array}{ll} \max\limits_{\mathbf{x}} & p\mathbf{1}^T\mathbf{x} - \mathbf{c}^T\mathbf{x} \\ \textit{subject to:} & \\ & \mathbf{g}(\mathbf{Q},\mathbf{x}) \leq \left[\mathbf{1}^T\mathbf{x}\right]\mathbf{s} \\ & \mathbf{H}\mathbf{x} \leq \boldsymbol{\rho} \end{array} \tag{3.4}$$

have some uncertainty associated with them. Starting with the objective function, where global and local market forces can cause fluctuations in the selling price of gasoline (p). Also, the cost data for blend feedstocks ($\mathbf{c}$) is imperfectly known because it is difficult to assign economic values to streams which are not sold directly as products by the refinery. Calculation of the cost incurred in producing these intermediates is non-trivial. This uncertainty in the slope of the objective (profit) function can cause the apparent (calculated) optimum to shift to a sub-optimal operating point. As seen in Figure 2.1, the calculated operating point though not violating any constraints, can result in the blender making a costlier final blend.

Parameters in the constraint functions in Problem 3.4 also have associated uncertainty. As seen in Chapter 2, uncertainty in the constraints can occur either in the constraint function parameters or, in the right hand side parameters. For the gasoline blending optimization problem, this translates to uncertain feedstock qualities and uncertain demands/availabilities respectively. The right hand side parameters in the quality specification constraints are the specifications themselves and the function parameters for the demand/availability constraints are integral coefficients. These are defined up front in the formulation of the problem and thus, are certain.

In the gasoline blend quality specification constraints, the feedstock qualities can be uncertain due to the following reasons:

1. The processes (FCCU, Reformer, Atmospheric Column, and so forth) the particular feedstock comes from are subject to continuous disturbances, which can be operation changes by the operators of unit, and/or stochastic (unmeasured) disturbances entering the process. In Figure 3.1, both the feedstock quality trajectory and the blend quality trajectory are shown as deviations from nominal feedstock and blend qualities. The stochastic disturbances affecting a feedstock production process can be from many different sources such as: *i)* composition changes in the crude oil being processed by the refinery, *ii)* ambient temperature affecting the operation of the separation columns (*e.g.*, main atmospheric

column, columns in separation trains of the FCCU, Reformer, *etc.*), *iii)* other process upsets in any of the upstream units. These disturbances become extremely important when running tanks are used for gasoline blending feedstocks or when blending is done in-line. During in-line blending, the blend feedstocks are fed straight into the blender manifold without being first routed through a tank. If running feedstock tanks are used, any disturbance in the feed qualities is filtered through the tank dynamics before it affects the blended gasoline qualities. In both cases, disturbances in feedstock qualities are transmitted through to the blended gasoline. Infrequent measurement of feedstock qualities (in some cases, the feedstock qualities are measured as infrequently as once a week) make the task of tracking these disturbances even more difficult.

2. Measurement errors from quality sensors.

Feedstock quality disturbances in the first category above can have the following disturbance structures (or a combination) : *i)* random steps (*e.g.*, crude switches and operator intervention), *ii)* coloured noise and random walk type disturbances, *iii)* cyclic disturbances (*e.g.*, day/night temperature cycling). In the second category (*i.e.*, measurement errors), the disturbances may be a combination of white noise and instrument biases.

Uncertainty in feedstock qualities implies uncertain parameters in the **Q** matrix (left hand side coefficients in the constraints) in Problem 3.4. As the shape of the feasible region is uncertain, nominal values for the uncertain parameters in the RTO problem can give solutions (blend recipes) which are infeasible (Figure 2.3) in the sense that implementing the recipe will cause violation of one or more quality specifications. In actual practise, when a blended batch is found to be off-specification it cannot be sold as scheduled. The off-specification blend has to be reblended with other feedstocks (usually the more expensive ones) or expensive additives to bring the batch up to specifications. The following costs can be associated with reblending [Grosdidier, 1997]: *i*) demurrage cost from having to hold the batch in inventory while the correction is done, *ii*) inventory cost from not having the money in the bank earning interest because the shipment is delayed, and *iii*) utilities cost, which is the cost of pump circulation required for mixing the batch after the correction is done. Another cost that can be associated with reblending is the cost of the feedstock(s) used for correction of the blend. However, these are relatively minor in magnitude compared to the indirect cost incurred due to the loss in the scheduled production rate.

In the demand and availability constraints, the demands and availabilities: ρ in

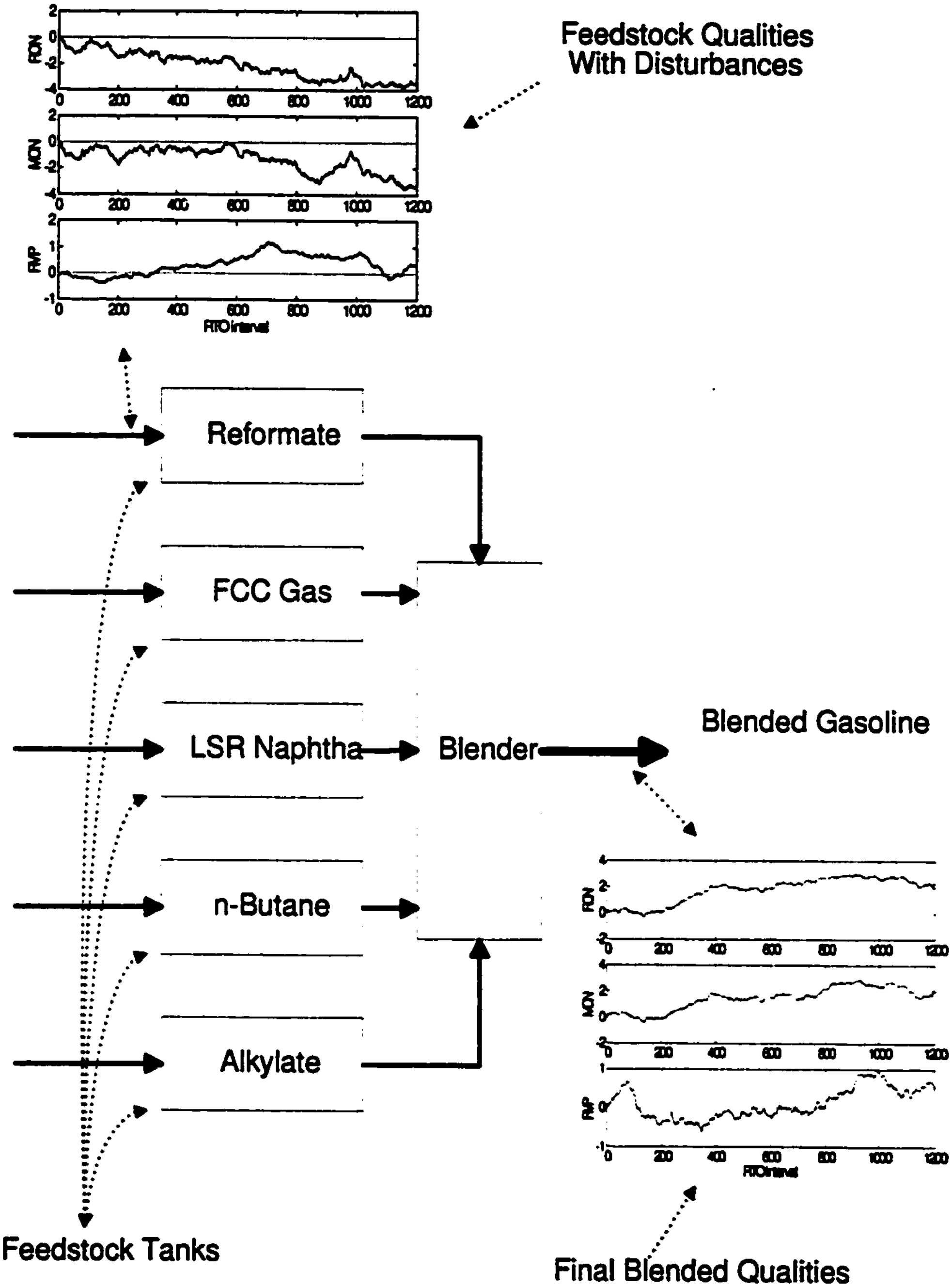

Figure 3.1: Propagation of feedstock quality disturbances through the gasoline blender

Problem 3.4 (right hand sides in the constraints), can only be known with some degree of uncertainty. Demand for any grade of gasoline is usually projected by the refinery's planning department from market data. Such forecasts always have some associated uncertainty. The blend feedstock availabilities are based on target production values set again by the planning department. There can be situations, however, where it is considered necessary to deviate from those production values. One such scenario could be that the price of a certain blend feedstock (or an intermediate) goes up in the market and the refinery decides to divert some of the scheduled production of this stream for direct selling on the market. Another situation with similar effects could be an emergency shutdown of one or more of the upstream units from which the blend feedstocks originate. Situations like the above cause uncertainty in blend feedstock availabilities. This again, can cause the generation of solutions which when implemented violate one or more of the demand/availability constraints and thus, are infeasible (Figure 2.2).

While the first type of uncertainty (uncertain cost data) can cause generation of sub-optimal solutions, where the solution makes less money than is possible, the second and third types (uncertainty in the constraints) can actually cause implementation of blend recipes which are infeasible. These infeasibilities could be due to violation of the quality specifications or the demand and availability constraints as seen above. This work concentrates on how to minimize constraint violation in the second category (*i.e.*, uncertain feedstock qualities leading to critical specification violations (see Figures 2.2 and 2.3)), while maximizing profit.

3.3 Blending Optimizers for Uncertain Feedstock Qualities

In this section, different blending optimizer formulations are developed that explicitly include uncertainty in feedstock qualities in their formulations. The Stochastic Programming techniques selected in Chapter 2 to be used in these RTO formulations are further explained before applying them in the blend optimizers.

3.3.1 GBO using Probabilistic Programming

The objective in gasoline blending is to produce the most economic blend while ensuring that the specification/demand/availability constraints are satisfied. As mentioned earlier, this is not always possible in the presence of uncertain gasoline feedstock qualities. Thus, Probabilistic Programming is a good candidate technique for use in the blending control problem as the risk of making an infeasible blend can be specified in the problem formulation (Problems 2.5 and 2.7).

Using Chance Constrained Programming (CCP), a minimum probability of feasibility can be specified for each uncertain constraint. In the blending problem, where the quality specification constraints (minimum *RON* and *MON* and maximum *RVP*) have uncertainty associated, individual chance constraints can be used with a high probability level to get "desirable" or low risk blend recipes. In other words, CCP can be used to ensure that most blends obtained are on-specification (quality constraints not violated, and thus no reblending required).

The chance constrained problem for gasoline blending is:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{e}^T\mathbf{x} \\ \textit{subject to:} & \\ & Prob[(\tilde{\mathbf{q}}_i - s_i\mathbf{1}^T)\mathbf{x} \le 0] \ge \pi_i \text{ for } i = 1, 2, ..., n_{qual} \\ & \mathbf{H}\mathbf{x} \le \boldsymbol{\rho} \end{array} \tag{3.5}$$

where: $\mathbf{e} = p\mathbf{1}^T - \mathbf{c}^T$, is the vector of economic data, $\tilde{\mathbf{q}}_i$ is the vector of uncertain feedstock quality indices for the i^{th} quality, s_i is the specification for the i^{th} quality in the final blend, n_{qual} is the number of uncertain quality constraints and, π_i is the minimum probability level at with which the i^{th} quality constraint must be satisfied. It turns out that there are two ways of converting the above problem to its certainty equivalent *i.e.*, an optimization problem with no uncertain parameters.

The chance constrained approach [Vajda, 1972] replaces the probabilistic constraints with non-linear constraints and is the classical chance constrained programming approach. For the case where the j^{th} feedstock quality's disturbance is modeled as a $N(0, \sigma_j^2)$ process, the chance constrained optimization problem becomes:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{e}^T\mathbf{x} \\ \textit{subject to:} & \\ & (s_i\mathbf{1}^T - \tilde{\mathbf{q}}_i)\mathbf{x} \ge -\tau_i\sqrt{\sum\limits_j \sigma_j^2 x_j^2} \\ & \mathbf{H}\mathbf{x} \le \boldsymbol{\rho} \end{array} \tag{3.6}$$

where τ_i is defined by:

$$\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\tau_i} e^{-\frac{x^2}{2}}dx = 1 - \pi_i \tag{3.7}$$

or

$$\tau_i = \sqrt{2}\, erf^{-1}(1 - 2\pi_i) \tag{3.8}$$

where erf is the error function. The right hand side of the first constraint can be interpreted as $-\tau_i \times$ *standard deviation of* i^{th} *constraint* (Appendix B).

A second method for solving the chance constrained LP [De Hennin, 1994], uses the estimated variance of the active constraints directly in calculating the back-offs from active constraints. The chance constrained LP with the set of active constraints known is converted to:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{c}^T\mathbf{x} \\ \textit{subject to:} & \\ & \mathbf{A}_a\mathbf{x} + \boldsymbol{\delta} = \mathbf{b}_a \end{array} \tag{3.9}$$

where $\mathbf{A}_a$ and $\mathbf{b}_a$ are the expected values of active constraint parameters and the constraint right hand sides respectively. $\boldsymbol{\delta}$ [deHennin, 1994] is the vector of back-offs from the active constraints required to satisfy the probabilistic constraints in the chance constrained program and is given as:

$$\delta_i = \sqrt{\Psi_{g,i,i}}\sqrt{2}\, erf^{-1}(2\pi_i - 1) \tag{3.10}$$

where $\Psi_{g,i,i}$ is the variance of the i^{th} active constraint function.

Unlike the Chance Constrained approach described in the previous subsection, the modified constraints (with back-offs) are linear. Thus, the nature of the original optimization problem is preserved and LP solvers can be used to solve the Probabilistic Programming problem. For the blending problem:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{e}^T\mathbf{x} \\ \textit{subject to:} & \\ & \left\{ \begin{array}{l} (\tilde{\mathbf{q}}_i - s_i\mathbf{1}^T)\mathbf{x} \leq 0 \text{ if } i \text{ inactive quality constraint} \\ (\tilde{\mathbf{q}}_i - s_i\mathbf{1}^T)\mathbf{x} + \sqrt{\Psi_{g,i,i}}\sqrt{2} erf^{-1}(2\pi_i - 1) = 0 \text{ if } i \text{ active constraint} \end{array} \right\} \\ & \mathbf{H}\mathbf{x} \leq \boldsymbol{\rho} \end{array} \tag{3.11}$$

where $\Psi_{g,i,i} = \sum\limits_j \sigma_j^2 x_j^2$ with σ_i as defined in the previous subsection.

A major assumption above was that the feedstock flows, solutions of the chance constrained LP, are already known. This is impossible in practise but, if the flows from the original solution of the LP with deterministic constraints (the expected values for the constraint parameters used to get the solution) are used as a first approximation and then iterated through the back-off problem, the flows (solution) for the chance constrained problem can be obtained. The computational cost of this iteration process needs to be taken into account while evaluating this approach. The other assumptions include that the active set (assumed known) is not altered by changes in feedstock qualities.

Comparing the above form with the Chance Constrained Program form (Problem 3.6), it can be seen that in the CCP form, implicit back-off terms are defined for all uncertain chance constraints. In the current approach, however, only the active constraints have the back-off term in them. Thus, if the active constraints are known beforehand, the back-off from active constraints method can be used. On the other hand, if the set of active constraints at the solution is not known, back-offs could be generated from all the uncertain constraints giving an over-conservative solution.

The joint probabilistic constraint programming problem for gasoline blending can be written as:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{e}^T\mathbf{x} \\ \textit{subject to:} & \\ & Prob[(\tilde{\mathbf{Q}}-\mathbf{s}\mathbf{1}^T)\mathbf{x} \leq 0] \geq \pi \\ & \mathbf{H}\mathbf{x} \leq \rho \end{array} \tag{3.12}$$

where: π is now the probability level at which the whole set of uncertain constraints must be satisfied.

The above formulation implies that $\pi \times 100\%$ or more of the solutions of Problem 3.12 should be feasible (no constraint violated with respect to the model used). Although this problem form is more desirable than individual probabilistic constraints (§2.2.3), solutions to this problem are much more computationally intensive. A conservative, approximate solution to Problem 3.12 can however, be obtained by using the probability bounding techniques outlined in Appendix C.

3.3.2 Probabilistic Programming for Non-Linear Blending Model

Probabilistic Programming has been discussed thus far only for linear models and constraint sets. As discussed in Chapter 1, it is generally accepted that non-linear blending models for Octane Number and Reid Vapour Pressure blending are more accurate and give more reproducible results. Unfortunately, there is no general method for solving non-linear chance constrained programs [Prekopa, 1995]. One possibility is to locally linearize the non-linear constraints and iteratively solve the CCP formulation for the linearized constraint set. This approach however, offers no convergence guarantees for the solution.

The non-linear quality constraints for the blending control problem can be obtained from the nonlinear RT-70 model, Equations 3.3 and 3.2. These constraints can be written in vector form as:

$$\begin{aligned} g_{RON}(\mathbf{r},\mathbf{m},\mathbf{o},\mathbf{b},\mathbf{x}) = \mathbf{r}^T\mathbf{x} + a_1[\mathbf{r}^T diag(\mathbf{r}-\mathbf{m})\mathbf{x} - (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{r}^T\mathbf{x}(\mathbf{r}-\mathbf{m})^T\mathbf{x}] + \ldots \\ a_2[\mathbf{o}_{sq}^T\mathbf{x} - (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{o}^T\mathbf{x}\mathbf{o}^T\mathbf{x}] + \ldots \\ a_3[\mathbf{b}_{sq}^T\mathbf{x} - (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{b}^T\mathbf{x}\mathbf{b}^T\mathbf{x}] \geq RON_{spec}(\mathbf{1}^T\mathbf{x}) \end{aligned} \tag{3.13}$$

$$\begin{aligned} g_{MON}(\mathbf{r},\mathbf{m},\mathbf{o},\mathbf{b},\mathbf{x}) = \mathbf{m}^T\mathbf{x} + a_4[\mathbf{m}^T diag(\mathbf{r}-\mathbf{m})\mathbf{x} - (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{m}^T\mathbf{x}(\mathbf{r}-\mathbf{m})^T\mathbf{x}] + \ldots \\ a_5[\mathbf{o}_{sq}^T\mathbf{x} - (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{o}^T\mathbf{x}\mathbf{o}^T\mathbf{x}] + \ldots \\ \widetilde{a}_6(\mathbf{1}^T\mathbf{x})^{-1}[\mathbf{b}_{sq}^T\mathbf{x} - (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{b}^T\mathbf{x}\mathbf{b}^T\mathbf{x}]^2 \geq MON_{spec}(\mathbf{1}^T\mathbf{x}) \end{aligned} \tag{3.14}$$

$$\mathbf{v}^T\mathbf{x} \leq RVPBI_{spec}(\mathbf{1}^T\mathbf{x}) \tag{3.15}$$

where: $\mathbf{r}$ and $\mathbf{m}$ are the vectors of *RON*s and *MON*s respectively for each feedstock, a_1, a_2, a_3, a_4, a_5, and $\widetilde{a}_6$ are the coefficients of the octane model ($\widetilde{a}_6 = \frac{a_6}{10000}$), $\mathbf{o}$ and $\mathbf{b}$ are the vectors of feedstock olefin and aromatic content respectively, $\mathbf{o}_{sq}$ and $\mathbf{b}_{sq}$ are vectors of squares of feedstock olefin and aromatic content, $\mathbf{v}$ is the vector of blending indices of the feedstock $RVPs$ and $\mathbf{x}$ is the vector of feedstock flowrates. After taking a first order Taylor series approximation of the above nonlinear constraint functions, the system of linearized octane number constraints obtained is:

$$\begin{aligned} \mathbf{g}_o + |\nabla_{\mathbf{x}}\mathbf{g}|_o(\mathbf{x}-\mathbf{x}_o) + \ldots \\ + |\nabla_{\mathbf{r}}\mathbf{g}|_o(\mathbf{r}-\mathbf{r}_o) + |\nabla_{\mathbf{m}}\mathbf{g}|_o(\mathbf{m}-\mathbf{m}_o) \geq \begin{bmatrix} RON_{spec} \\ MON_{spec} \end{bmatrix} (\mathbf{1}^T\mathbf{x}) \end{aligned} \tag{3.16}$$

where: $\mathbf{r}_o$, $\mathbf{m}_o$, $\mathbf{x}_o$ are the nominal $RONs$, $MONs$ and feedstock flowrates, and $\nabla_{\mathbf{x}}\mathbf{g}$, $\nabla_{\mathbf{r}}\mathbf{g}$ and $\nabla_{\mathbf{m}}\mathbf{g}$ are the first partial derivatives of the non-linear functions in the Ethyl model as defined in Equations 3.13 and 3.14.

The above set of linearized *RON* and *MON* constraints (3.16) along with the maximum *RVP* constraint gives the set of specification constraints and can be written as:

$$[\mathbf{Q}_a - \mathbf{s}\mathbf{1}^T]\mathbf{x} \leq \boldsymbol{\ell} \tag{3.17}$$

where: $\mathbf{Q}_a$ is the augmented quality matrix (Equation 3.18) and $\boldsymbol{\ell}$ (Equation 3.19) contains all the other terms from the linearization that are either constants (nominal operating point values) or have uncertain feedstock *RON* and *MON* values.

$$\mathbf{Q}_a = \begin{bmatrix} -\left|\nabla_{\mathbf{x}}\mathbf{g}\right|_o \\ \mathbf{v}^T \end{bmatrix} \tag{3.18}$$

$$\boldsymbol{\ell} = \begin{bmatrix} \mathbf{g}_o - \left|\nabla_{\mathbf{x}}\mathbf{g}\right|_o \mathbf{x}_o + \left|\nabla_{\mathbf{r}}\mathbf{g}\right|_o (\mathbf{r} - \mathbf{r}_o) + \left|\nabla_{\mathbf{m}}\mathbf{g}\right|_o (\mathbf{m} - \mathbf{m}_o) \\ 0 \end{bmatrix} \tag{3.19}$$

Using these linearized constraints, the static Gasoline Blending Control problem using CCP becomes:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \mathbf{e}^T\mathbf{x} \\ \textit{subject to:} & \\ & Prob[(\mathbf{q}_{a,i} - s_i\mathbf{1}^T)\mathbf{x} \leq \ell_i] \geq \pi_i \text{ for } i = 1, 2, \ldots, n_{qual} \\ & \mathbf{H}\mathbf{x} \leq \boldsymbol{\rho} \end{array} \tag{3.20}$$

where: $\mathbf{q}_{a,i}$ is the i^{th} row of the augmented quality matrix (Equation 3.18) and ℓ_i is the i^{th} element of the $\boldsymbol{\ell}$ vector (Equation 3.19).

The only uncertain quantities in Problem 3.20 are in the elements ℓ_i as the vectors $\mathbf{q}_{a,i}$ are comprised of the nominal quality vectors $\mathbf{r}_o$ and $\mathbf{m}_o$. The above problem formulation is static in nature in the sense that it does not take into account the time variation of the feedstock qualities. It is based on the current nominal values for the feedstock qualities and only guarantees that the current section of the blend shall satisfy the constraints at a high probability level. It does not project into the future or look back at what has been blended so far. As discussed in § 1.3.3, the Time-Horizon formulation for the blend optimizer proposed in [Singh, 1997] is designed to overcome this shortcoming. In the next section, methods to incorporate uncertainty information into variations of the Time-Horizon optimizer are described.

3.3.3 Probabilistic Programming for Dynamic Blending

As described in Singh [1997], Time-Horizon RTO based blend optimization consists of predicting the blend feedstock qualities over the future and minimizing blending cost over the period of the blending. Both the quality specification as well as demand constraints are framed such that they are satisfied at the end of the blend. A variation of the above approach would be to have the flexibility to tune the feasibility horizon. In the flexible feasibility horizon approach (referred to as the Feasibility Horizon RTO method in the rest of this document), the optimization problem is formulated such that the blend quality constraints are required to meet blend quality specifications at the end of the number of intervals given by the feasibility horizon. As in the full THRTO method, the past blended gasoline qualities and quantities (starting from the beginning of the blend) are considered in the optimization. Unlike the THRTO method however, feedstock flowrates are only calculated for the number of future RTO intervals specified by the feasibility horizon. With the assumption that some information is given about the measurement noise and the feedstock qualities' prediction uncertainties, Probabilistic Programming can be used to solve Dynamic Blending problems. In this section, the Probabilistic Programming approach is extended to the cases of gasoline blend optimization over a time horizon.

THRTO (feasibility over the blend)

As seen in §1.3.3, the THRTO algorithm proposed by Singh [1997], predicts feedstock qualities over the next few optimization intervals and uses these predictions to optimize over the remaining portion of the blend. The future feedstock quality predictions enter the problem as constraint parameters and thus, probabilistic constraints can be used to incorporate the uncertainty in these parameters into the blend optimization problem. Also, the current measured feedstock qualities will always have some uncertainty associated which can also be handled using probabilistic constraints. The resulting formulations for Dynamic Blend Optimization using Probabilistic Programming are given below.

For the linear blending model:

$$\begin{aligned} &\min_{\mathbf{x}} \quad \sum_{t=k}^{n} \mathbf{e}^T \mathbf{x}_t \\ &\textit{subject to:} \\ &\quad Prob\left[\sum_{t=1}^{k-1} \bar{\mathbf{Q}}_t \mathbf{x}_t + \sum_{t=k}^{n} \hat{\hat{\mathbf{Q}}}_t \mathbf{x}_t \leq \mathbf{s} \sum_{t=1}^{n} \left(\mathbf{1}^T \mathbf{x}\right)\right] \geq \pi \\ &\quad \sum_{t=1}^{n} \mathbf{H} \mathbf{x}_t \leq \rho \end{aligned} \tag{3.21}$$

where: $\mathbf{e}$ is the vector of economic data, $\mathbf{x}_t$ is the vector of feedstock flowrates during the t^{th} optimization interval, $\bar{\mathbf{Q}}_t$ is the matrix of feedstock quality indices during past intervals and, $\hat{\hat{\mathbf{Q}}}_t$ is the matrix of feedstock quality indices during the t^{th} optimization interval in the future. Also, k is the present RTO interval and n is the last RTO interval (end of blend).

It has been assumed that the feedstock qualities remain constant between RTO intervals (slow enough dynamics). The summation terms over the unknown optimization variables (current and future feedstock flowrates) can be removed from the constraints in Problem 3.21 to get:

$$\begin{aligned} &\min_{\mathbf{x}_{k2n}} \quad \mathbf{e}_{k2n}^T \mathbf{x}_{k2n} \\ &\textit{subject to:} \\ &\quad Prob\left[\left(\hat{\hat{\mathbf{Q}}}_{k2n} - \mathbf{s}\mathbf{1}_{k2n}^T\right) \mathbf{x}_{k2n} \leq -\sum_{t=1}^{k-1} \left(\bar{\mathbf{Q}}_t - \mathbf{s}\mathbf{1}^T\right) \mathbf{x}_t\right] \geq \pi \\ &\quad \mathbf{H}_{k2n} \mathbf{x}_{k2n} \leq -\sum_{t=1}^{k-1} \mathbf{H} \mathbf{x}_t + \rho \end{aligned} \tag{3.22}$$

where: $\mathbf{e}_{k2n}^T = \left[\ \mathbf{e}^T \quad \mathbf{e}^T \quad \dots \quad \mathbf{e}^T\ \right]$, is the economic vector $\mathbf{e}^T$ appended row-wise, once for each RTO interval (from k to n), $\hat{\mathbf{Q}}_{k2n} = \left[\hat{\mathbf{Q}}_k \quad \hat{\mathbf{Q}}_{k+1} \quad \dots \quad \hat{\mathbf{Q}}_n\right]$ is the matrix of predicted present and future feed quality indices, $\mathbf{H}_{k2n} = [\mathbf{H}_k \quad \mathbf{H}_{k+1} \quad \dots \quad \mathbf{H}_n]$ is the matrix for demand and availability constraints, $\mathbf{x}_{k2n} = \left[\ \mathbf{x}_k^T \quad \mathbf{x}_{k+1}^T \quad \dots \quad \mathbf{x}_n^T\ \right]^T$ is the vector of present and future feed flows, and $\mathbf{1}_{k2n}$ is a vector (size: $(n-k+1) \times number\ of\ feedstocks \times number\ of\ grades$) whose elements are ones.

In Problem 3.22, the terms containing the unknown optimization variables are on the left hand side in the inequality constraints. The structure of this formulation is that of the simple Chance Constrained Program, Problem 2.17, and given uncertainty information for the predicted feedstock qualities ($\hat{\hat{\mathbf{Q}}}_{k2n}$), the deterministic equivalent of this formulation can be obtained. In Time-Horizon Blend Optimization, the size of the optimization problem changes at each RTO interval k, and the number of optimization variables (feedstock flows to be calculated) is given by the product of the number of feedstocks available, the number of different gasoline grades being blended, and the number of RTO intervals left:

$$N_{opt\_variables} = N_{feedstocks} \times N_{grades} \times (n - k + 1) \tag{3.23}$$

For all the development in this chapter it is assumed that only one grade of gasoline is being blended. The formulations for more than one grade follow easily from this development and are described in Chapter 4 with the case-study description.

Instead of using linear octane blending models with blend indices, the non-linear model (Equations 3.3) can be linearized around the current operating point (Appendix D) and used in the formulation of the Time-Horizon Blend Optimization problem. This formulation can be written as:

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \sum\limits_{t=k}^{n} \mathbf{e}^T \mathbf{x}_k \\ \textit{subject to:} & \\ & Prob\left[\sum\limits_{t=1}^{k-1} \mathbf{Q}_{a_t}\mathbf{x}_t + \sum\limits_{t=k}^{n} \widehat{\mathbf{Q}}_{a_t}\mathbf{x}_t \leq \sum\limits_{t=1}^{k-1} \left\{\mathbf{s}\left(\mathbf{1}^T\mathbf{x}_t\right) + \ell_t\right\} + \sum\limits_{t=k}^{n} \left\{\mathbf{s}\left(\mathbf{1}^T\mathbf{x}_t\right) + \widehat{\ell}_t\right\}\right] \geq \pi \\ & \sum\limits_{t=1}^{n} \mathbf{H}\mathbf{x}_t \leq \rho \end{array} \tag{3.24}$$

or

$$\begin{array}{ll} \min\limits_{\mathbf{x}} & \sum\limits_{t=k}^{n} \mathbf{e}^T \mathbf{x}_k \\ \textit{subject to:} & \\ & Prob\left[\sum\limits_{t=k}^{n} \left(\widehat{\mathbf{Q}}_{a_t} - \mathbf{s}\mathbf{1}^T\right)\mathbf{x}_t \leq -\sum\limits_{t=1}^{k-1} \mathbf{Q}_{a_t}\mathbf{x}_t + \sum\limits_{t=1}^{k-1} \left\{\mathbf{s}\left(\mathbf{1}^T\mathbf{x}_t\right) + \ell_t\right\} + \sum\limits_{t=k}^{n} \widehat{\ell}_t\right] \geq \pi \\ & \sum\limits_{t=k}^{n} \mathbf{H}\mathbf{x}_t \leq -\sum\limits_{t=1}^{k-1} \mathbf{H}\mathbf{x}_t + \rho \end{array} \tag{3.25}$$

where: $\mathbf{Q}_{a_t}(\widehat{\mathbf{Q}}_{a_t})$ and $\ell_t(\widehat{\ell}_t)$ are described by Equations 3.18 and 3.19 respectively for the t^{th} RTO interval. The carats denote predicted quantities.

Problem 3.25 is slightly different from the Chance Constrained Programs discussed up until this point because it has uncertain parameters in the right hand side vector of the quality specification inequality constraints. The uncertainty in these terms can easily be incorporated when the deterministic equivalent of this CCP formulation is developed (using the methods described in Chapter 2). If all the uncertain parameters in a Chance Constrained Program are in right hand side vector, the deterministic equivalent of that CCP is a Linear Program (LP). Thus, even with explicit inclusion of uncertainty into linear constraints with uncertain right hand sides, those constraints remain linear in the decision parameters, $\mathbf{x}$ [Charnes and Cooper, 1963].

Feasibility Horizon RTO

The Feasibility Horizon RTO (FHRTO) approach defines the time horizon over which the optimization is done, to be from the present RTO interval to the feasibility horizon or the end of the blend batch (whichever is closer). The feasibility horizon can vary from zero (feasible over current interval) to the number of intervals in a blend batch (feasible over blend). This formulation of the Feasibility Horizon RTO problem is given by:

$$
\begin{array}{ll}
\min\limits_{\mathbf{x}} & \sum\limits_{t=k}^{m} \mathbf{e}^T \mathbf{x}_t \\
\textit{subject to:} & \\
& \sum\limits_{t=1}^{k-1} \bar{\mathbf{Q}}_t \mathbf{x}_t + \sum\limits_{t=k}^{m} \hat{\mathbf{Q}}_t \mathbf{x}_t \leq \mathbf{s} \sum\limits_{t=1}^{m} (\mathbf{1}^T \mathbf{x}) \\
& \mathbf{H}_t \mathbf{x}_t \leq \rho
\end{array}
\tag{3.26}
$$

where: m is the lesser of the feasibility horizon and the last RTO interval (end of blend). In Model Predictive Control terms, the prediction horizon is the lesser of the feasibility horizon and the end of the blend batch. The feedstock qualities are predicted for the feasibility horizon and these predictions as well as the past blended gasoline qualities are used to calculate optimal feedstock flowrates over the feasibility horizon. Also, the demand and availability constraints have to be framed differently than for the original THRTO approach. This is because, although the actual product demands are over the length of the blend, the formulation below can only manipulate feedstock flowrates over the feasibility horizon. One possible solution is to split the product demand over the length of the blend and specify limits for each subbatch (RTO interval). While the resulting demand constraints are simple enough to formulate, they do not allow the optimizer the greater flexibility given by demand constraints over the blend, to maneuver allocated blend amount in each RTO interval. This is how the demand constraints are formed for the Feasibility Horizon RTO based algorithms in Chapter 4.

The Chance Constrained form of the Feasibility Horizon RTO problem (Problem 3.27) is framed such that the current RTO solution ensures that the blend produced by mixing the calculated quantities of feedstocks with the blend produced so far, satisfies each quality specification at a minimum probability level of π.

$$
\begin{array}{ll}
\min\limits_{\mathbf{x}} & \sum\limits_{t=k}^{m} \mathbf{e}^T \mathbf{x}_t \\
\textit{subject to:} & \\
& Prob\left[\sum\limits_{t=1}^{k-1} \bar{\mathbf{Q}}_t \mathbf{x}_t + \sum\limits_{t=k}^{m} \hat{\hat{\mathbf{Q}}}_t \mathbf{x}_t \leq \mathbf{s} \sum\limits_{t=1}^{m} \left(\mathbf{1}^T \mathbf{x}\right)\right] \geq \pi \\
& \mathbf{H}_t \mathbf{x}_t \leq \rho
\end{array}
\tag{3.27}
$$

The problem size, or number of optimization variables in Problems 3.26 and 3.27, is given by the following:

$$
N_{opt\_variables} = (m - k + 1) \times N_{feedstocks} \times N_{grades} \tag{3.28}
$$

One of the advantages of using the FHRTO Interval formulation is smaller optimization problem size, and thus, faster execution time. On the other hand, the solutions obtained are more conservative as the formulation requires that the quality specifications be satisfied earlier in the blend (instead of just at the end of the blend).

3.4 Summary

Different formulations designed to handle uncertainty in Gasoline Blending quality constraints have been developed in this chapter. Variations on the Time-Horizon RTO formulation [Singh, 1997] are extended to explicitly include parameter uncertainty in the quality specification constraints. In the next chapter, these forms are compared against conventional Gasoline Blending Optimizers by using a simple case study.

Chapter 4

Blending Optimization Case Study

The techniques developed in Chapter 3 to counter the ill-effects of uncertainty in gasoline blending optimization quality constraints need to be benchmarked against conventional blending optimization formulations. The linear blending models described in §3.1.1, the blending octane number model for *RON* and *MON*, and the blending indices model for *RVP*, are used in this case study. This combination of blending models is used for both simulating the blending process, as well as for framing the quality constraints in all the blending optimization formulations. The non-linear Ethyl RT-70 models described in §3.1.2 are not used because the models are non-associative (Appendix E). One of the consequences of this non-associativity of the Ethyl models is that the blended qualities are different for each different sequence of blending a fixed set of feedstocks (the quantities and qualities of the feedstocks are fixed). As the same models are used in the optimization as in the simulation, there is no structural mismatch between the actual process and the model of the process used for optimization.

To get a better understanding of the performance of the different RTO schemes, a benchmark that represents the best possible blender performance is needed. In this work, the best possible blender performance is taken to be that obtained when, the blending optimization is done with all blend parameters known exactly over the period of the blend [Singh, 1997]. For the case of linear blending models, this reduces to the solution of :

$$\begin{array}{ll} \min\limits_{\mathbf{x}_{t_o \Rightarrow t_f}} & \mathbf{e}^T_{t_o \Rightarrow t_f} \mathbf{x}_{t_o \Rightarrow t_f} \\ \textit{subject to:} & \\ & \widetilde{\mathbf{Q}}_{t_o \Rightarrow t_f} \mathbf{x}_{t_o \Rightarrow t_f} \leq \mathbf{s}\left(\mathbf{1}^T \mathbf{x}_{t_o \Rightarrow t_f}\right) \\ & \mathbf{H}\mathbf{x}_{t_o \Rightarrow t_f} \leq \boldsymbol{\rho} \end{array} \tag{4.1}$$

where: $\mathbf{e}_{t_o \Rightarrow t_f}$ is the vector of economic data. $\mathbf{x}_{t_o \Rightarrow t_f} = \left[\ \mathbf{x}^T_{t_o}\ \ \mathbf{x}^T_{t_o+\Delta t}\ \cdots\ \mathbf{x}^T_{t_f}\ \right]^T$ is the vector of all feedstock flows over the blend, where $\mathbf{x}_t$ is the vector of feedstock flowrates during the optimization interval ($t \Rightarrow t+\Delta t$). $\widetilde{\mathbf{Q}}_{t_o \Rightarrow t_f} = \left[\widetilde{\mathbf{Q}}_{t_o} \; \widetilde{\mathbf{Q}}_{t_o+\Delta t} \; \cdots \; \widetilde{\mathbf{Q}}_{t_f} \right]$ where $\widetilde{\mathbf{Q}}_t$ is the matrix of feedstock quality indices during the above interval ($\widetilde{\mathbf{Q}}_t$ assumed known for all t, present and future). t_o is the start time of the blend while t_f is the end time of the blend. $\mathbf{H}$ is the coefficient matrix for the demand and availability constraints, and $\boldsymbol{\rho}$ is the vector of blend demands and feedstock availabilities.

The negative of the objective function value at the solution of Problem 4.1 gives the maximum achievable profit for a blend. The profit obtained from using different RTO schemes, when compared to this upper limit gives a useful measure of the ability of the RTO scheme to reduce the effect of parametric disturbances on optimality.

4.1 Case Study Description

The benchmark problem is based on the case studies in [Singh, 1997] in which, both regular and premium gasolines are to be blended simultaneously. The number of feedstocks is limited to five and the number of blends to two (Figure 4.1). The feedstocks to the blender are :

1. Gasoline fraction from the Catalytic Reformer or Reformate.
2. Gasoline fraction from the Fluidized Catalytic Cracking Unit (FCCU Gas).
3. Light Straight Run Naphtha from the Atmospheric Tower (LSR Naphtha).
4. Pure n-Butane from various units.
5. Gasoline fraction from the Alkylation Unit (Alkylate).

For the simulation, there are two blenders working in parallel, one for regular grade gasoline and the other for premium grade gasoline. Both blenders produce batches over a twenty four (24) hour period. The blended batches have minimum and maximum demand constraints (Table 4.1). Each feedstock has associated maximum availability constraints (Table 4.2). It is assumed that all the feedstocks are coming to the blender through running (§1.3.5 and §3.2), constant volume, feedstock tanks. The demand and availability constraints are applied on each individual RTO interval instead of over the complete 24 hour batch. The benchmark case study consists of blending one hundred (100) batches of both grades of gasoline to compare the performance of conventional blending optimizers to that of the proposed optimizers. Each 24 hour batch has twelve (12) RTO intervals.

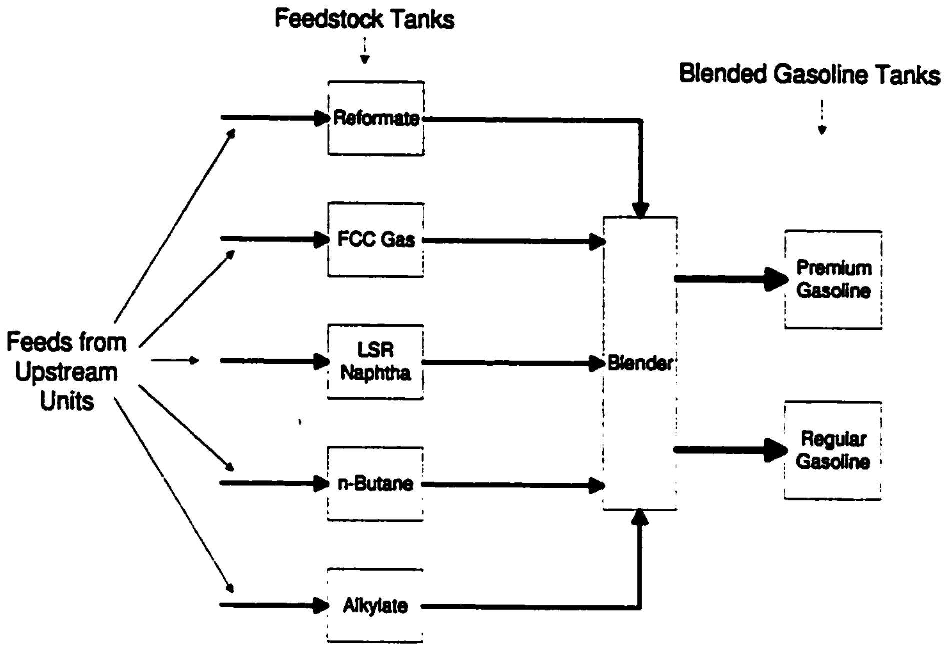

Figure 4.1: Gasoline Blending Flow Diagram

	Premium	*Regular*
Maximum demand, bbl/day	11000	9000
Minimum demand, bbl/day	11000	9000
Minimum RON	91.5	88.5
Minimum MON	80.0	77.0
Maximum RVP	10.8	10.8
Value, $/bbl	37.50	33.00

Table 4.1: Blended Gasoline Requirements

Feedstock	**Cost ($/bbl)**	**Availability (bbl/day)**
Reformate	34.00	8000
LSR Naphtha	26.00	6500
n – Butane	10.30	4000
FCC Gas	31.30	6000
Alkylate	37.00	5000

Table 4.2: Feedstock Costs and Availabilities

Feedstock	RON	MON	RVP
Reformate	94.1	80.5	3.8
LSR Naphtha	70.7	68.7	12.0
n – Butane	93.8	90.0	51.5
FCC Gas	92.9	80.8	5.3
Alkylate	95.0	91.7	6.6

Table 4.3: Feedstock Qualities

Quality	$\sigma_{disturbance}$	$\sigma_{measurement\_noise}$
RON	0.3	0.3
MON	0.3	0.3
RVP	0.15	0.15

Table 4.4: Standard deviations of the uncertain feedstock qualities and associated measurement noise with the feedstock quality sensors

The given feedstock qualities (Table 4.3) will be taken as the mean or expected feedstock qualities in all the cases studied. The actual feedstock properties are modeled as non-stationary disturbances superimposed on the nominal (constant) values. In all the RTO structures considered, the RTO is executed every 2 hours, and although optimal feedstock flow-rates are generated for the remaining blend, they are only executed for the current RTO interval. It is also assumed that the execution time of the optimizer is negligible. The structure of the noise at the input to the running tanks is that of a random walk sequence or integrated white noise [Box and Jenkins, 1976]. The transfer function representation for integrated white noise is:

$$x_k = \frac{1}{1 - z^{-1}} u_k \tag{4.2}$$

where u is zero mean, white noise (standard deviations for each quality given in Table 4.4) , and x_k is the deviation (at the entrance to the running tank) of the actual feedstock quality from the nominal value at the k^{th} RTO interval.

The tanks themselves are assumed to be well mixed and to hold a constant level at all times. The transfer function representation for all feedstock tanks is:

$$y_k = \frac{0.39 z^{-1}}{1 - 0.61 z^{-1}} x_k \tag{4.3}$$

where y_k is the deviation (at the entrance to the blender) of the actual feedstock quality from the nominal value at the k^{th} RTO interval. The transfer function for all the feedstock tanks (Equation 4.3) is a first order filter with a time constant of four (4) hours. y_k is applied as the disturbance to the feedstock quality at RTO interval k. Thus, the quality disturbance at each feedstock inlet to the blender is white noise coming through an integrator and a well mixed, constant level tank.

In addition to the above mentioned disturbances, all quality sensors (of all feedstocks) have associated measurement noise. The measurement noise is assumed to have a zero mean, white noise structure. The disturbance term is assumed to remain constant during each RTO interval (two *hours*). The overall transfer functions for the actual and measured feedstock qualities are thus given by:

$$\begin{aligned} y_k &= \frac{0.39z^{-1}}{(1-0.61z^{-1})(1-z^{-1})}u_k \\ \widehat{y}_k &= \frac{0.39z^{-1}}{(1-0.61z^{-1})(1-z^{-1})}u_k + w_k \end{aligned} \tag{4.4}$$

where $\widehat{y}_k$ is the deviation of the measured feedstock quality from the nominal value and w_k is the measurement noise term with standard deviations given in the second column of Table 4.4.

The two octane numbers are related measures of the same property for gasoline viz., the ability to resist pre-ignition combustion [Palmer and Smith, 1985]. This translates to a strong correlation between the *RON* and *MON* for any gasoline feedstock or blend. Therefore, the quality disturbances (y_k) are designed such that the *RON* and *MON* of each feedstock are correlated with a correlation coefficient of approximately 0.9. The *RVP* of a gasoline is a property that may be unrelated to the octane numbers. Thus, the correlation between the *RVP* of a feedstock and *RON* or *MON* of the same feedstock is statistically insignificant in this case-study.

As the nature of the disturbances is stochastic, a long enough simulation is needed so that "good" results are obtained, in a statistical sense. For the above reason, all the optimizers are run on a set of one hundred (100), twenty four (24) hour blend simulations. Thus, the benchmarking case study consists of one hundred, day long blends produced using the various blending optimization algorithms. The RTO frequency (as mentioned above) is fixed at 12 day^{-1}. This makes all the individual quality disturbance vectors 1200 elements long. The same disturbance vectors (different for each feedstock and quality) are used in all the case-studies.

Optimizer performance is measured by the profit generated by the particular optimizer over one hundred, one day simulations. In the case where the optimizer generates blends which are feasible (in the sense that they meet all the specifications), profit calculation is easy as the cost data is known. If the blend at the end

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,618.15	11,000.00
Average daily profit ($/day)	29,858.10	73,874.86

Table 4.5: Results for the THRTO (Ideal case) algorithm

of the day is off-specification, profit calculations become more difficult because the off-specification blend has to be reblended before it can be sold. For the above calculation, reblending costs are needed which as discussed in § 3.2 are not easy to determine. The procedure adopted here is that the profit is reduced by 10% for the days where the final blend is infeasible. If the measured actual qualities at the end of a day do not meet product specifications, that day's profit is calculated as:

$$profit_{infeasible\ blend} = 0.9 \times (price\ of\ blend\ produced - cost\ of\ feedstocks\ used)$$

While calculating the number of infeasible blends and thus the profit for batches penalized with reblending costs, an allowance is made for the maximum violation permitted before the batch is marked as infeasible. A violation of blended *RON* and *MON* by 0.03 octane number and 0.005 *psi* for the blended RVP is deemed acceptable.

All numerical computation for the case study, including the blending simulation and optimization was done using **Matlab 5.3** [The Mathworks, 1999*a*] and the **Optimization Toolbox 2.0** [The Mathworks, 1999*b*]. The hardware platform was an Intel® Celeron™ 366 MHz machine with 96 MB RAM.

4.2 Results

The RTO algorithms from Chapters 1 and 3 were implemented on the data-set described above (Section § 4.1). In this section, the results generated using the various RTO algorithms are compared against those given by Problem 4.1.

4.2.1 Time Horizon RTO - Ideal case

Solving 4.1 for the feedstock flowrates with the blend demands and feed availabilities given in Tables 4.1 and 4.2 generates the results given in Table 4.5.

The average daily profit obtained by applying the THRTO algorithm with all feedstock qualities known exactly is, $ 103,732.96 per day. As mentioned earlier, this numeric serves as a common benchmark which is used to compare the different algorithms. The blended qualities and the amount of gasoline blended at the end of each day's batch is plotted for both grades blended (Figures 4.2 and 4.3). As specified in Problem 4.1, these results are obtained by solving one optimization problem for each day's blend (instead of solving twelve optimization problems, one every two hours), with all feedstock qualities assumed known exactly. It can be seen in Figures 4.2 and 4.3 that both *RON* and *RVP* for regular grade as well as premium follow the specifications exactly. On the other hand, the blend *MON* is well above the minimum *MON* constraint for both grades. Studying the Lagrange multipliers for the blend quality specification constraints (Appendix G) verifies that the minimum *RON* and maximum *RVP* constraints are active for all 100 batches while the MON constraint is never pushed.

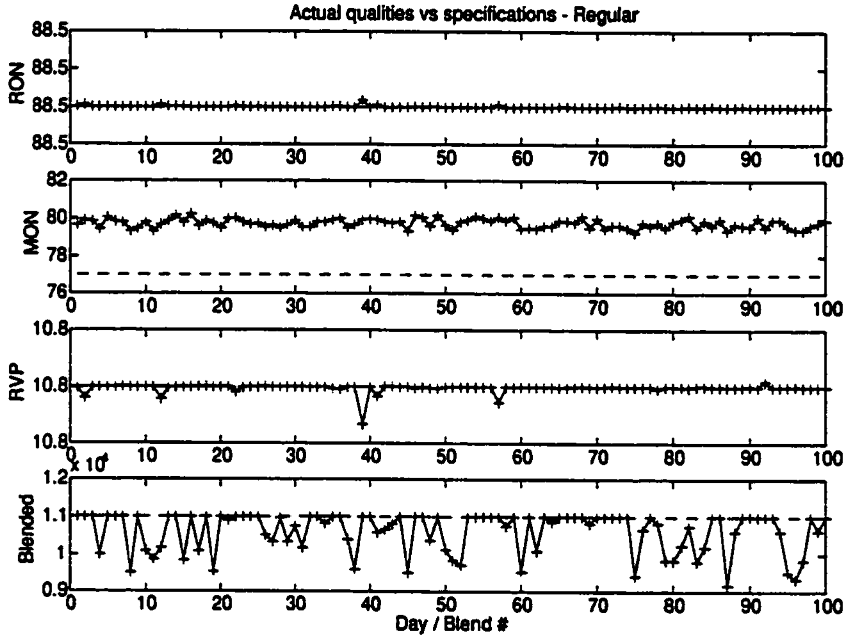

Figure 4.2: Results for benchmark algorithm (all qualities known): regular

The demand constraints used throughout the case study were modified from minimum and maximum "blend batch" demand constraints (applied over the whole day's batch) to demand constraints over each individual "sub-batch" (RTO interval). This is because, in the *LP + bias* and Feasibility Horizon RTO formulations, the blend period being considered might not include the full blend batch, whereas all feedstock flows to the end of the blend are needed to form "blend batch" constraints.

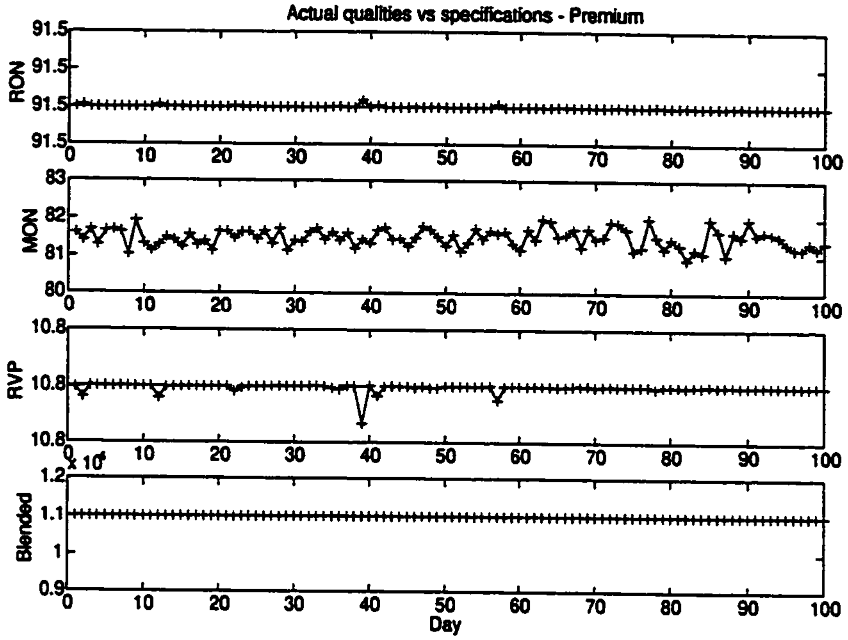

Figure 4.3: Results for benchmark (all qualities known) case: premium

4.2.2 Linear Programming & Bias Updating

The simplest formulation of RTO, Linear Programming with bias updates was used to generate blend recipes for the case study described in § 4.1. In the optimization step, only the current feed flowrates are calculated and as described in § 1.3, the LP + bias algorithm ignores what has been blended so far, as well as any predicted parameters when calculating feedstock flowrates for the current RTO interval. As the LP + bias approach optimizes over the current RTO interval only, the size of the optimization problem is given by:

$$N_{opt\_variables} = N_{feedstocks} \times N_{grades}. \tag{4.5}$$

Within the LP + bias framework there are two possible approaches that can be used for the model update phase. If there are no measurements available for the uncertain parameters in the blending model, nominal or expected values for these parameters (feedstock qualities in our case study) are used and the bias update step is expected to keep the model current. If, however, measurements for these parameters are available, these can be used to update the $\widetilde{Q}$ matrix in Problem 1.2 in addition to the bias updating (bias model updating is retained as the measurements used are noisy).

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	11,000.00	11,000.00
Average daily profit ($/day)	30,212.00	67,588.79
Infeasible blends (days)	70	67

Table 4.6: Results for the LP + bias updating (nominal feed qualities) algorithm

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,513.33	11,000.00
Average daily profit ($/day)	29,214.82	70,105.99
Infeasible blends (days)	38	36

Table 4.7: Results for the LP + bias updating measured feed qualities) algorithm

Both these variations to the LP + bias updating algorithm were implemented on the case study and the results are given in the sections that follow.

LP + bias updating using nominal feed qualities

In the first approach, only blended qualities are measured (after the fact). Nominal values are used for the feedstock qualities as it is assumed that they are not measured/measurable. The measured blended qualities from the previous RTO period/interval are used to calculate the bias (equation 1.3) which is updated. The results obtained by implementing this approach on the case study are given in Table 4.6.

The blended qualities and the amount of gasoline blended at the end of each day's batch is plotted for both grades blended (Figures 4.4 and 4.5). The blended RON and RVP are off-specification for more than half the (regular and premium) batches produced whereas the MON is within specifications for all one hundred days/batches. Thus, even though all batches for both blends are produced at the maximum allowed production rate (11000 bbl/day), there is a significant penalty imposed by reblending costs.

LP + bias updating using measured feed qualities

In this approach, measured feedstock qualities are used to update the blending model and the measured blended qualities are used in bias updating. The results in Table 4.7 were obtained by using this scheme of model updating.

Using this modified LP + bias updating approach, significantly better results are obtained with an increase of more than $ 1500/*day* in the average daily profit compared to the nominal LP + bias updating approach. This increased profitability

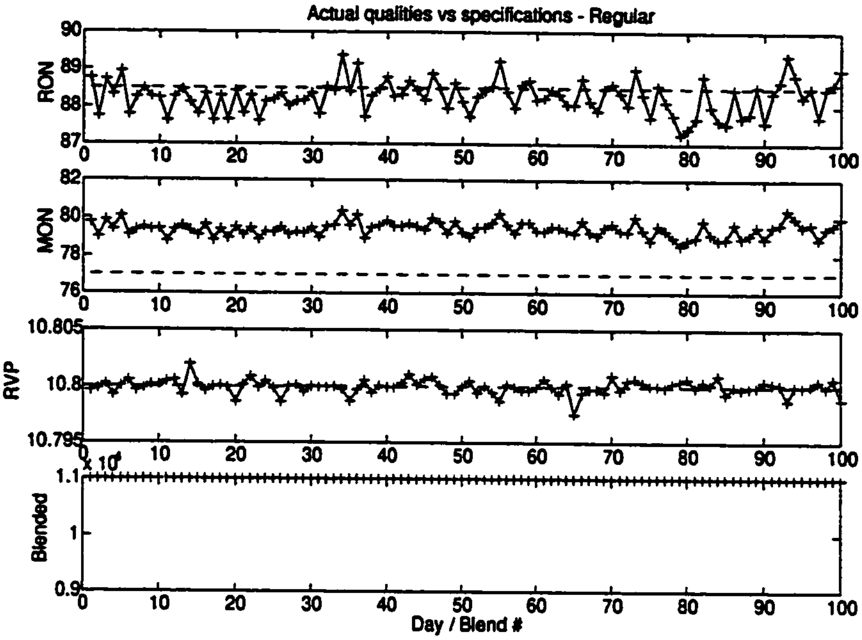

Figure 4.4: Results for LP + bias updating with nominal feedstock qualities: regular

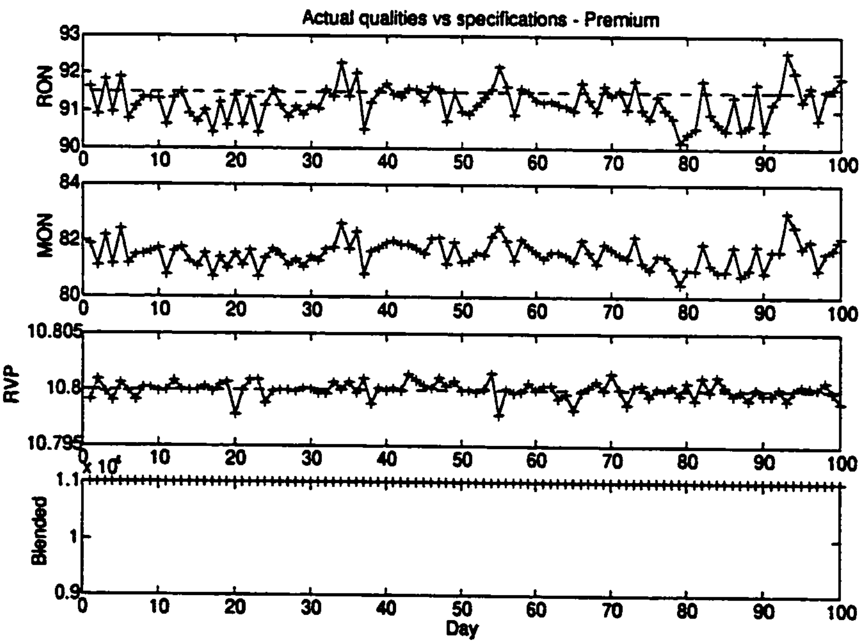

Figure 4.5: Results for LP + bias updating with nominal feedstock qualities: premium

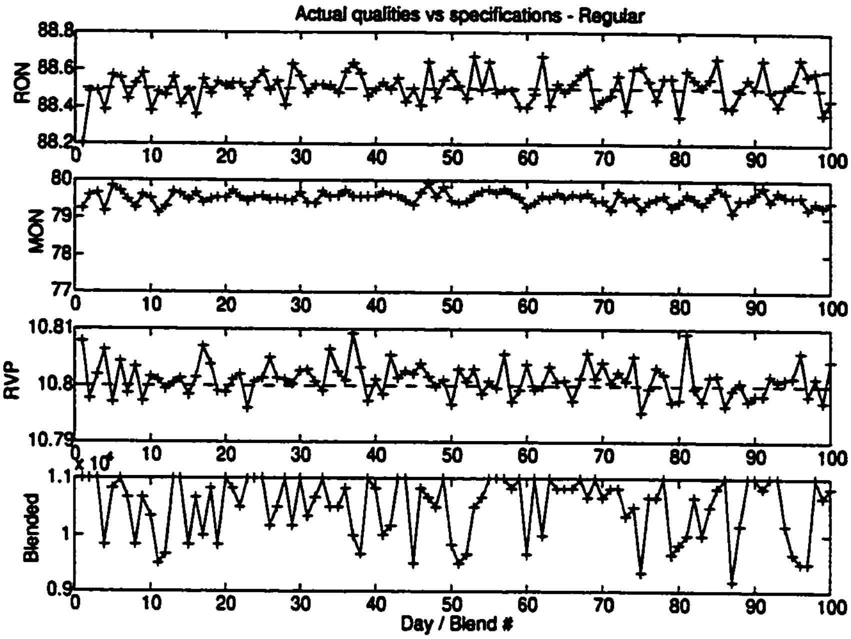

Figure 4.6: LP + bias updating using measured feed qualities: regular grade

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,297.94	10,796.17
Average daily profit ($/day)	29,608.54	69,557.99
Infeasible blends (days)	24	26

Table 4.8: Results for the THRTO (deterministic constraints) algorithm

is due to the decrease in the production of off-specification blends and inspite of the somewhat decreased regular blend production. The plots of the blended qualities (Figures 4.6 and 4.7) also show a tighter control on *RON* and *RVP*.

4.2.3 Time-Horizon RTO - Deterministic constraints

The Time-Horizon RTO approach (§ 1.3) uses current measured feed qualities, predicted future feed qualities and measured qualities for the blend produced thus far to calculate blend recipes over the remaining portion of the current batch. This formulation looks at the whole batch at each RTO execution to generate an optimal trajectory for the feedstock flowrates. Even though feedstock flowrates are calculated for all future intervals at each step, only the recipe for the current interval is implemented. This process is repeated until the end of the batch is reached. The results obtained by applying the THRTO formulation on the case-study are presented in Table 4.8.

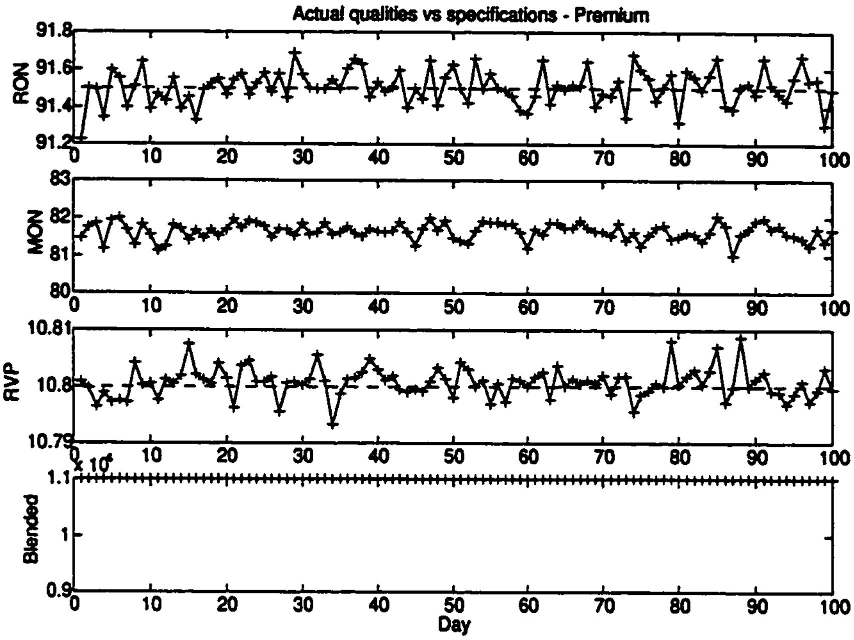

Figure 4.7: LP + bias updating with measured feedstock qualities: premium grade

From the blended quality and quantity plots in Figures 4.8 and 4.9, some features are observed which were not present in the blend quality plots studied thus far. Although there are less infeasible blends produced with this approach than using either of the LP with bias updating formulations, the infeasible blends produced using the blend over horizon form have relatively larger specification violations (for both *RON* and *RVP*). Examination of the optimization results for the sub-batches (results of each RTO step) for the batches/days with the infeasible blends, reveals that most of the infeasible blends result from optimization problems that have no feasible solution. Thus, in these cases, at some RTO interval before the blend is complete, the optimization problem to be solved to obtain blend recipes becomes infeasible. As a result of the above, most batches with blend quality specification violations also violate the minimum required production/demand constraints (Figures 4.8 and 4.9). This reduction in production resulted in the lower total profit seen for this formulation.

4.2.4 Time-Horizon RTO - Chance constraints

The chance constrained time horizon RTO formulation is derived from the time horizon RTO form by using probabilistic constraints in place of the deterministic quality specification constraints (§ 3.3.3). The probabilistic or chance constraints formed use

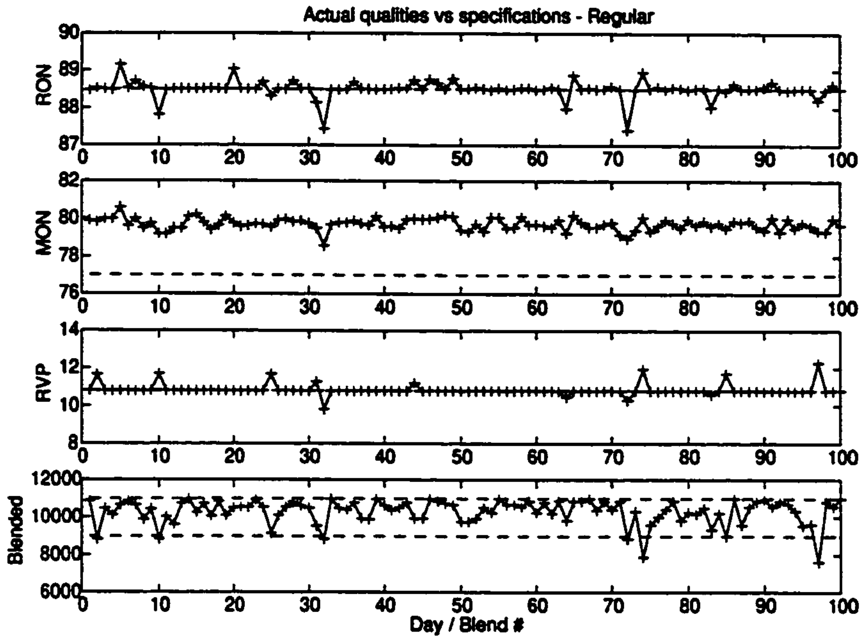

Figure 4.8: Results from Time-Horizon RTO algorithm: regular

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,375.57	10,916.99
Average daily profit ($/day)	29,153.70	72,448.41
Infeasible blends (days)	8	7

Table 4.9: Results for the THRTO (chance constraints) algorithm

the quality disturbance structure information and try to ensure that the probability of the quality constraints being satisfied at the end of the blend is high. For the current case-study, the chance constraints use the statistics of both the future feedstock quality predictions as well as the measurement noise associated with the feed quality sensors. Individual chance constraints are used with the minimum probability of constraint satisfaction set at 95%. Thus, in Problem 3.22, $\pi = \left[\begin{array}{ccc} 0.95 & 0.95 & 0.95 \end{array} \right]^T$. The results obtained by using this formulation are given below along with the plots of blended qualities and quantities (Figures 4.10 and 4.11) for both gasoline grades.

Results for the chance constrained THRTO approach exhibit behaviour similar to the THRTO results described above in § 4.2.3. However, the number of infeasible blends produced is reduced to less than a third for a total of 15 infeasible blends. This along with increased production rates give an average daily profit that is $ 2,130.84 less than the maximum achievable average daily profit.

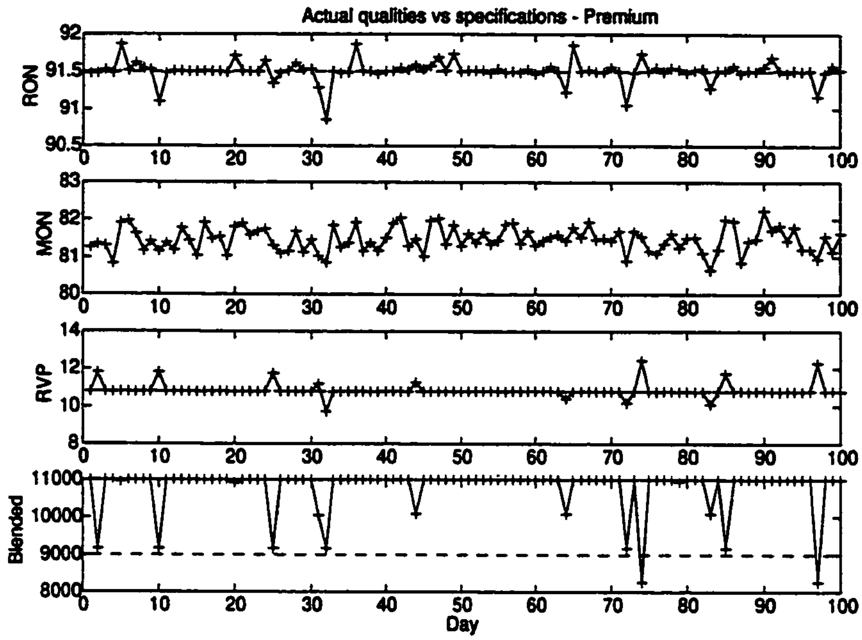

Figure 4.9: Results from THRTO algorithm: premium

4.2.5 Feasibility Horizon RTO - Deterministic constraints

The Feasibility Horizon RTO (FHRTO) formulation is a dynamic blending optimization formulation derived from the THRTO form (§ 3.3.3). The optimization problem at each interval is done over a flexible, receding horizon instead of over the complete blend. Although results were generated for $m \in [k, k+1, k+2, k+3]$ in Problem 3.26, only the zero-step ahead ($m = k$) Feasibility Horizon RTO or the Feasible over Current Interval results are presented here. Results for the one, two, and three step ahead FHRTO forms applied to the case study are summarized in Appendix H.

Feasible over Current Interval

In the zero-step ahead Feasibility Horizon RTO or Feasible over Current Interval approach, qualities of the blend produced up to the current RTO interval are used along with measurements of current feedstock qualities to calculate the blending recipe for the current RTO interval. As feedstock flow-rates are calculated only for the current RTO interval, the quality constraints specify that the blend be feasible at the end of the current RTO execution instead of the end of the batch/day. The results obtained using the feasible over current interval RTO scheme are given in Table 4.10.

The feasible over current interval approach applied to the case study results in a relatively high total profit partly due to the low number of infeasible batches produced.

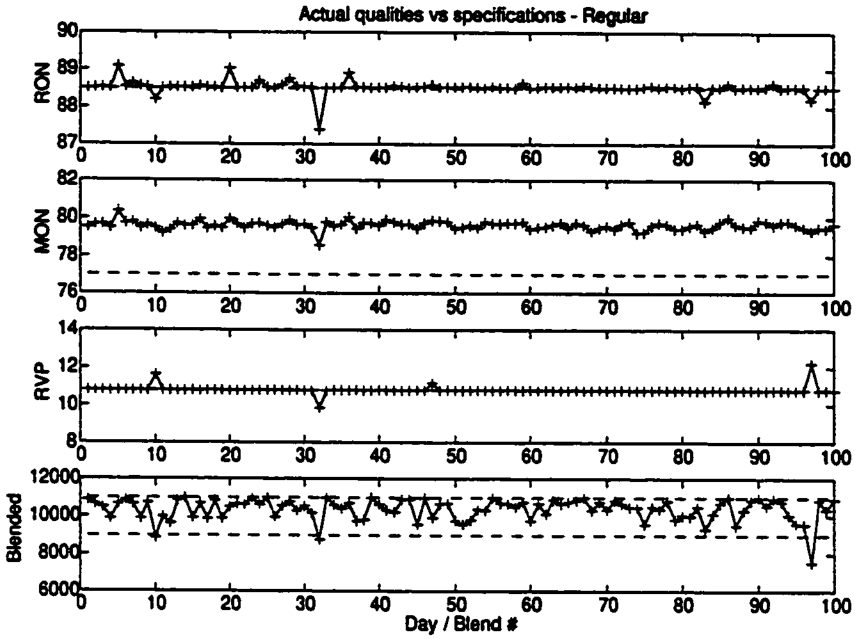

Figure 4.10: Results from chance constrained THRTO algorithm: regular

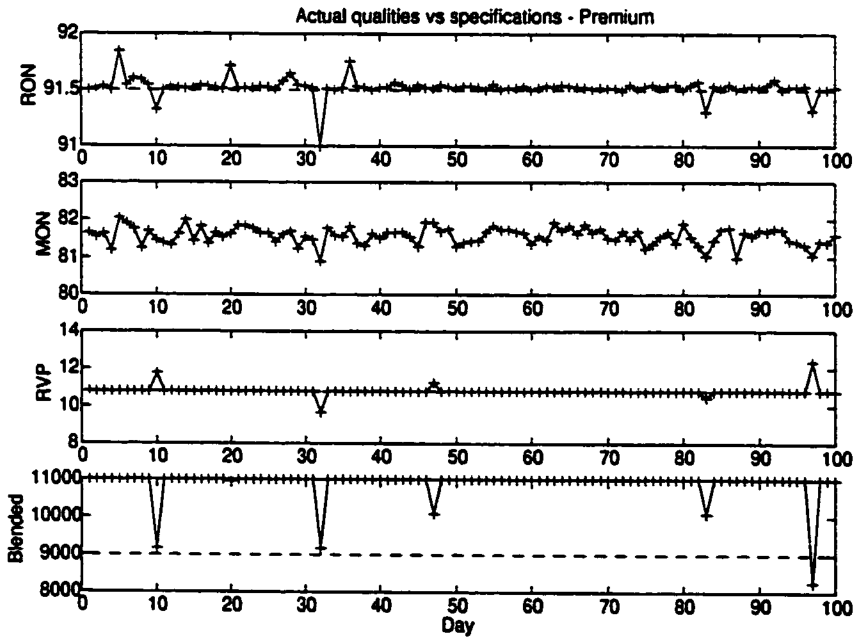

Figure 4.11: Results from chance constrained THRTO algorithm: premium

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,513.33	11,000.00
Average daily profit ($/day)	29,622.23	71,623.98
Infeasible blends (days)	17	19

Table 4.10: Results for the FNRTO (deterministic constraints) algorithm

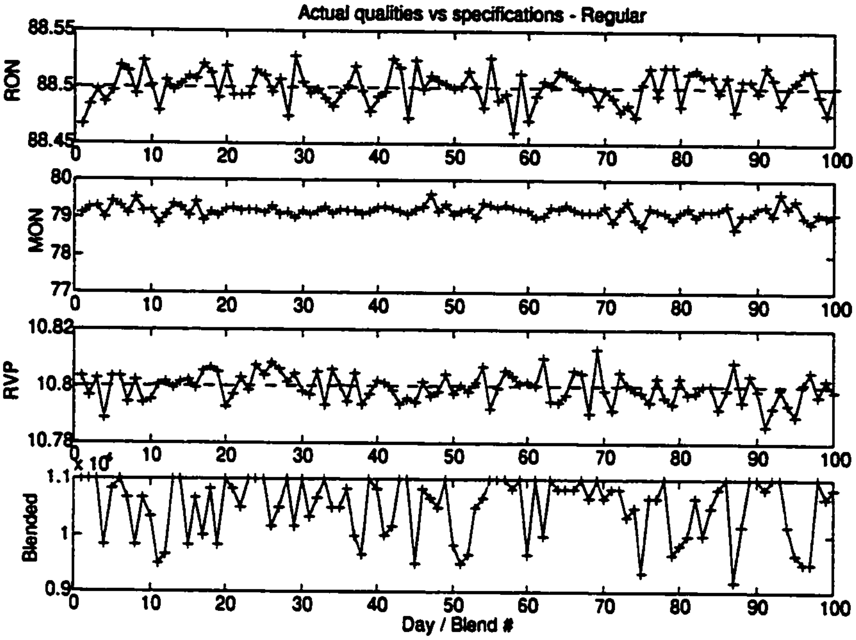

Figure 4.12: Feasible over Current Interval RTO algorithm: regular

This is also apparent from the quality and productions plots in Figures 4.12 and 4.13 which show the tight spread of the blended *RON* and *RVP* variation. The premium blend production rate plot indicates that it stayed at the demand maximum constraint throughout the 100 day long batches.

4.2.6 Feasibility Horizon RTO - Chance constraints

The chance constrained Feasibility Horizon RTO formulation extends the Feasibility Horizon RTO scheme to include chance or probabilistic constraints for quality specification constraints that use measured current feedstock qualities as well as predicted future feed qualities. Although results were generated for the zero-step ahead up to three-step ahead forms of Problem 3.27, only the zero-step ahead Feasibility Horizon RTO or the Feasible over Current Interval results are presented here. Results for the one, two, and three step ahead Chance Constrained FHRTO forms applied to the case study are summarized in Appendix H.

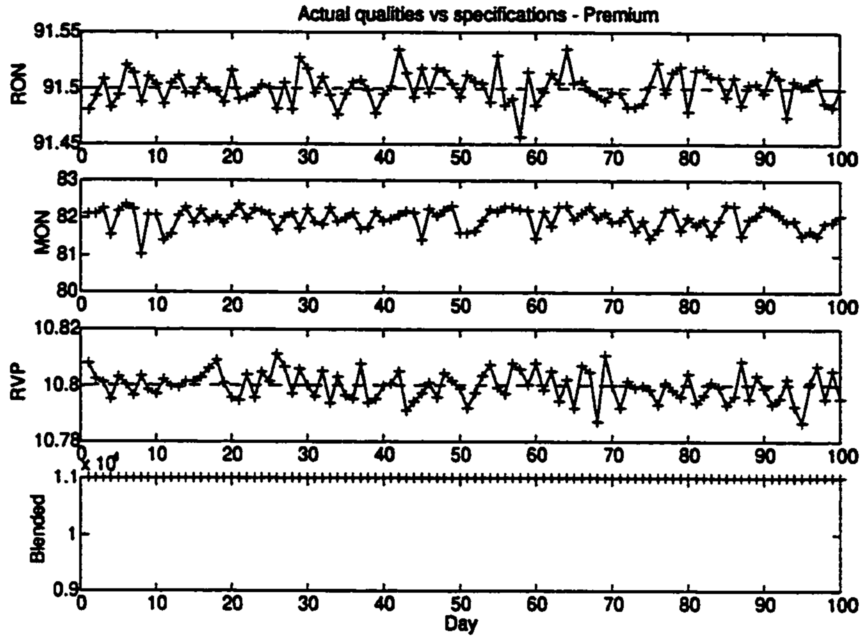

Figure 4.13: Feasible over Current Interval RTO algorithm: premium

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,367.19	11,000.00
Average daily profit ($/day)	29,999.16	72,751.01
Infeasible blends (days)	1	2

Table 4.11: Results for 0-step ahead FHRTO (chance constraints)

Feasible over Current Interval

In the 0-step ahead Feasibility Horizon RTO form ($m = k$ in Problem 3.27) only the current intervals feedstock flowrates are variables in the optimization problem. Thus, the only uncertainty in the formulation of the quality constraints comes from measurement noise associated with the measured current feedstock qualities.

The 0-step ahead FHRTO + CCP formulation results (Table 4.11) are the best in the case-study so far. The infeasible batches produced in the 100 day simulation are very few, and the overall daily profit high, as the reblending penalty for infeasible batches is avoided. Thus, even though the average production rate is lower than that for the Feasible over Current Interval case (without chance constraints), the average daily profit is higher by more than 1,500 $/day.

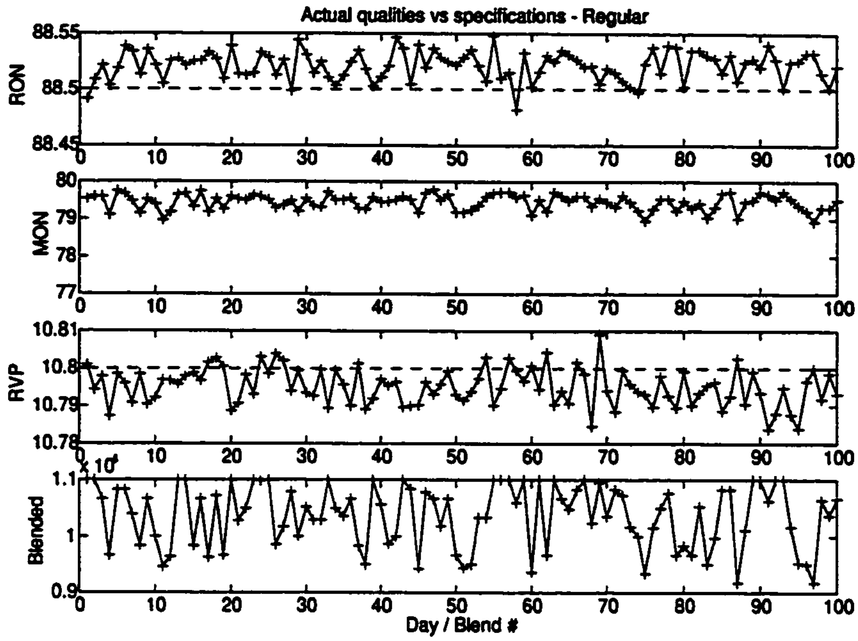

Figure 4.14: Results using chance constrained 0-step ahead FHRTO algorithm: regular

4.3 Discussions

In this chapter, different Real-Time Optimization algorithms were implemented on a simple gasoline blending case study to observe how these algorithms handle constraint parameter uncertainty. This uncertainty was simulated in the case study by using noisy measurements of disturbances in feedstock qualities being routed to the blender through running feedstock tanks. The three categories of RTO algorithms tested on the case study were: 1) conventional, static $LP + bias$ optimization algorithms; 2) dynamic Time-Horizon RTO algorithms with deterministic constraints, and; 3) dynamic Time-Horizon RTO algorithms with probabilistic constraints.

The $LP + bias$ algorithms only consider the current interval and do not consider the blend produced thus far or that to be produced over the blend horizon. As seen in Table 4.12, these factors contribute to the low average daily profit seen for the $LP + bias$ forms. The results for the $LP + bias$ algorithm that uses measured feedstock qualities (in addition to bias updating) are markedly better than those where the nominal feedstock qualities are used.

The Time-Horizon RTO algorithm is a special case of the Flexible Time-Horizon RTO scheme discussed in § 3.3.3 ($m = n$ in Problem 3.26). Results for this full Time-Horizon RTO algorithm also show a low average daily profit; comparable to

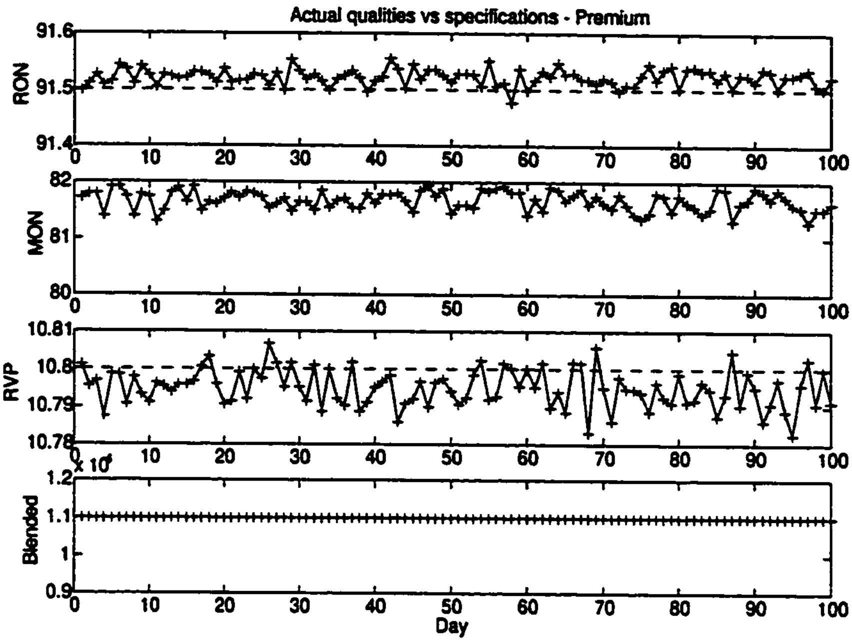

Figure 4.15: Results using chance constrained 0-step ahead FHRTO algorithm: premium

that obtained by using the *LP + bias* algorithm using measured feed qualities, even though the number of infeasible blends produced is reduced by a third. The full Time-Horizon RTO approach allows the optimizer some flexibility to sacrifice short term feasibility for overall economic gain. In some cases, this strategy can lead to a situation from which there are no feasible solutions. It is these situations that lead to the incomplete blends seen in § 4.2.3 where the optimizer itself fails to find a feasible solution at an RTO interval before the end of the blend. By introducing the feasibility horizon as a tunable parameter, the Feasibility Horizon RTO formulation offers an opportunity to trade-off between aiming for higher profitability and of avoiding the infeasibility problems in the full THRTO approach. Using a shorter feasibility horizon forces the optimizer to follow a more conservative solution trajectory. This is because there is less flexibility to trade-off short term feasibility for economic gain over the blend horizon.

Results for the zero-step ahead Feasibility Horizon RTO (or Feasible over Current Interval) approach reflect the above trade-off where short term feasibility is chosen over (possible) long term profitability. As the optimizer is required to generate solutions that satisfy quality constraints over the current RTO interval itself, the results for this approach show a significant improvement over the full Time-Horizon RTO

Algorithm	Profit ($/day)	$\frac{\text{Profit}_{algorithm}}{\text{Profit}_{ideal\ case}}$	Infeasible blends
$LP + bias$ (*nominal*)	$97,801	94.28%	137
$LP + bias$ (*measured*)	$99,321	95.75%	74
$THRTO$	$99,167	95.60%	50
$THRTO + CCP$	$101,602	97.95%	15
$0 - step\ FHRTO$	$101,246	97.60%	36
$1 - step\ FHRTO$	$101,309	97.66%	35
$2 - step\ FHRTO$	$99,933	96.34%	48
$3 - step\ FHRTO$	$98,957	95.40%	55
$0 - step\ FHRTO + CCP$	$102,750	99.05%	3
$1 - step\ FHRTO + CCP$	$102,565	98.87%	6
$2 - step\ FHRTO + CCP$	$102,378	98.69%	8
$3 - step\ FHRTO + CCP$	$102,116	98.44%	10

Table 4.12: Results for all RTO formulations in the gasoline blending case study

approach with less infeasible blends. Results for the one, two, and three step FHRTO forms, presented in Appendix H, seem to follow a trend where as the feasibility horizon grows from the current interval to three intervals ahead of the current interval, the profitability decreases. This trend may be a consequence of how the case study problem (the profitability measure in particular) is formulated and needs further investigation.

Results for the chance constrained form of the full THRTO (Problem 3.22) approach show an increase in average daily profit over the THRTO formulation with deterministic constraints. This improvement is expected as the probabilistic constraints for the blend quality specifications in the Time-Horizon RTO forms allow the Chance Constrained THRTO formulations to compensate for feedstock quality uncertainty directly. Significant improvements, especially in reduction of the number of total infeasible blends produced, were noted for each of the chance constrained forms of the Feasibility Horizon RTO formulations. The results for the chance constrained one, two, and three step ahead Feasibility Horizon RTO forms also exhibit a trend similar to that seen for their deterministic constraint forms with a decrease in average daily profit (and increase in number of infeasible blends) as the feasibility horizon grows from the current interval to three intervals ahead of the current interval.

The best performance was observed for the chance constrained form of the Feasible over Current Interval with only three infeasible blends. Examination of the blended quality profiles of the deterministic constrained and chance constrained zero-step ahead THRTO formulations (Figures 4.12, 4.13, 4.14, and 4.15) shows how the probabilistic constraints force the optimizer to back off from all the constraints that have uncertain parameters. On comparing the blended regular grade quality profiles for the above formulations in Figures 4.12 and 4.14, it can be observed that all the quality profiles are very similar between the two figures and both the octane number profiles in Figure 4.14 appear shifted upwards with respect to Figure 4.12. Similarly, the blended *RVP* profile is shifted downwards in Figure 4.14 with respect to Figure 4.12.

The results for the Time-Horizon RTO (both with and without chance constraints) presented in this chapter were generated using minimum and maximum demand constraints over each RTO interval. Allowing demand over the full blend constraints gives the optimizer additional flexibility/maneuverability to aim for higher economic benefit. Additional results were generated by using demand over blend constraints (along with relaxed individual interval demand constraints) for the deterministic as well as chance constrained THRTO forms. These results, along with a modified benchmark given by the ideal case Time-Horizon formulation, are presented in Appendix I. On comparing these results to those presented in this chapter, it can be seen that although the new benchmark shows an increased average daily profit (by $\sim$ \$218/day), the THRTO results using the demand over blend constraints are in fact worse than the results obtained using only demand over interval constraints. This behaviour seems to be another instance where additional flexibility to trade-off near term feasibility for increased profit over the blend leads to the optimizer getting into an unrecoverable situation.

The concept of quality give-away where blended qualities exceed specifications is given a lot of importance in current petroleum refining practice [Rigby and Warren, 1995; Honeywell Hi-Spec Solutions, viewed 24 September, 2000; Grosdidier, 1997]. A number of conventional gasoline blending optimizers formulate the optimization problem such that any quality give-away is minimized [Anonymous, 1997]. The primary reason behind this practice is that give-away is considered expensive because most of the feedstocks with desirable properties are more expensive. For example, Reformate, which in the case study has high Octane Numbers and an exceptionally low *RVP*, is one of the most expensive feedstocks. Minimizing quality give-away explicitly in the optimization formulation however, can be counter-productive, as it alters the objective function by reducing the importance of purely economic factors. The above statement is supported by the fact that all the blended quality profiles, for all the optimization algorithms used (including the benchmarks), show that blended *MON* exceeds blend grade specifications by a fair margin.

The economic performance of the different RTO approaches used in this case study is compared graphically in Figures 4.16, 4.17, and 4.18. These plots present

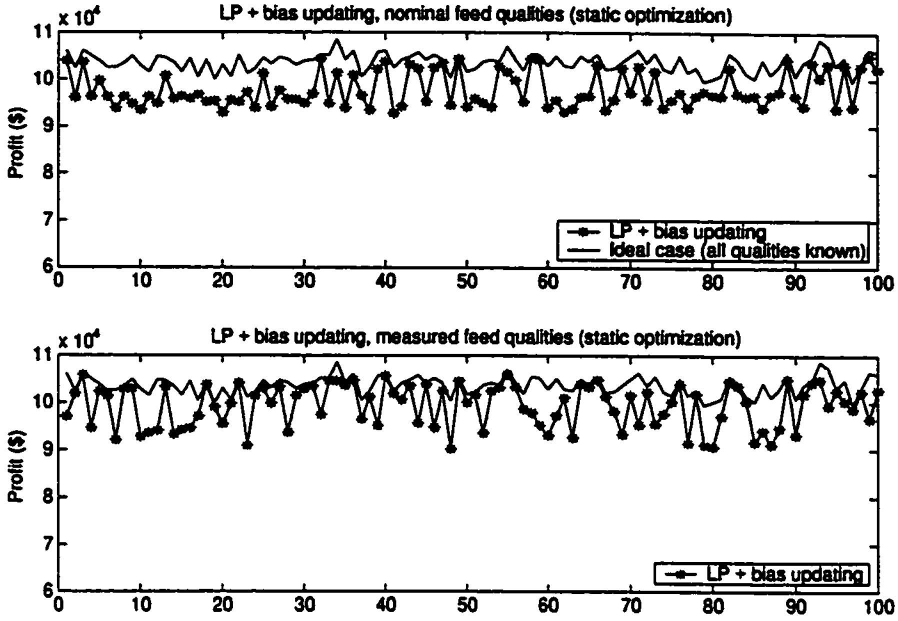

Figure 4.16: Performance (profit/blend) comparison between LP + bias updating algorithms and the benchmark algorithm

daily profit profiles for each of the Real-Time Optimizers along with the daily profit profile for the benchmark (all feedstock qualities known) THRTO ideal case. The daily profit profile for the benchmark (the upper line in all plots, with dots to represent daily profit) represents the maximum achievable profit on each of the 100 days in the case study. Thus, the area between these profit profiles represents lost profit. The ability of Probabilistic Programming formulations to reject the negative effects of uncertainty in constraint parameters is apparent on comparing the two plots in Figures 4.17, and 4.18.

4.3.1 Computational Issues

Worst case computation times for a single RTO interval are presented in Table 4.13 for all RTO algorithms used. As all deterministic constraints are linear in this case study, both of the LP + bias updating algorithms, and both of the Time Horizon RTO schemes translate to LP formulations. The Optimization Toolbox [Coleman *et al.*, 1999] uses an interior point method based on the predictor-corrector algorithm presented in Mehrotra, 1992 to solve the LP problems posed by the first four algorithms in Table 4.13. The RTO formulations with chance constraints (algorithms five and six in Table 4.13) result in optimization problems with non-linear constraints (§3.3.1). These NLP problems are solved using a sequential quadratic programming

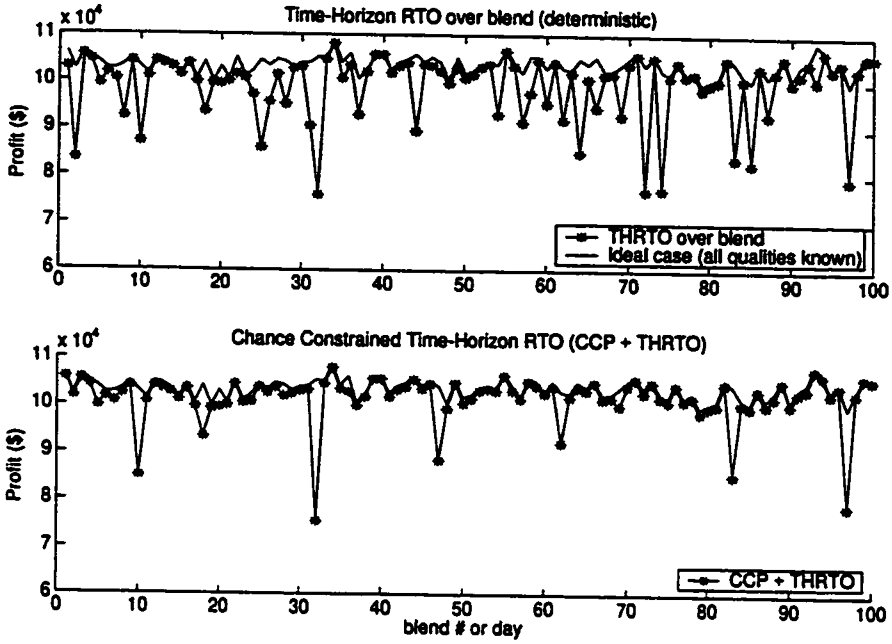

Figure 4.17: Profit profiles for THRTO algorithms

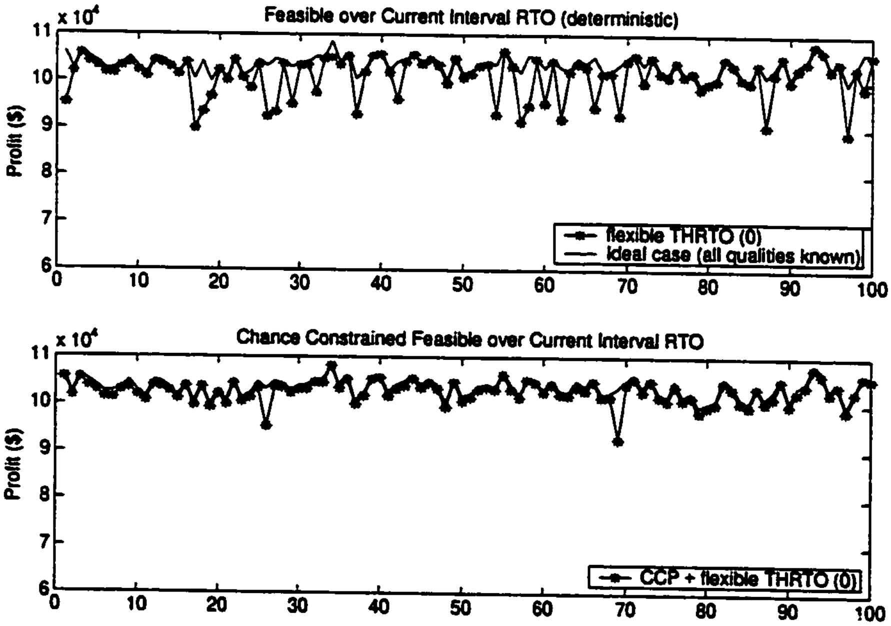

Figure 4.18: Profit profiles for Feasible over Current Interval algorithms

Algorithm	Worst case computation times
$LP + bias\ (nominal)$	0.8 seconds
$LP + bias\ (measured)$	0.8 seconds
$THRTO$	1.2 seconds
$0 - step\ FHRTO$	0.7 seconds
$THRTO + CCP$	11 minutes, 56.3 seconds
$0 - step\ FHRTO + CCP$	2.0 seconds

Table 4.13: Worst case computation times for the RTO formulations used

(SQP) algorithm by the Optimization Toolbox.

All RTO formulations used in the case study except for the two Time Horizon RTO (deterministic and chance constrained) require the solution of a fixed size optimization problem for all RTO intervals. This is because, both of the LP + bias, as well as the two Feasibility Horizon RTO forms only optimize for the current interval's feedstock flowrates, resulting in an optimization problem of size 10 ($N_{feedstocks} = 5$, $N_{grades} = 2$ in Equations 4.5 and 3.28). In the two Time Horizon RTO forms, the size of the optimization problem solved at each RTO interval in a blend batch varies from 120 to 10 ($N_{feedstocks} = 5$, $N_{grades} = 2$, while N_{RTO} varies from 12 to 1 in Equation 3.23) over the period of the blend.

The worst case single RTO computation times are very close for both LP + bias algorithms and the Feasibility Horizon RTO (deterministic constraints) algorithm because, for these three algorithms, each RTO computation involves the solution of a 10 optimization variable LP. The chance constrained Feasibility Horizon RTO formulation requires solution of a 10 optimization variable NLP problem, which accounts for the increase in computation time. For the Time Horizon RTO formulations, the largest computation times were required for the earlier RTO intervals when the number of optimization variables is higher. As seen in Table 4.13, the Chance Constrained THRTO (NLP) formulation required significantly more computation than the deterministic constrained THRTO (LP).

Chapter 5

Conclusions and Future Work

One of the guiding principles behind modern industry has always been achieving maximum possible profitability. In recent times, other principles, such as safety and environmental impact, have gained equal importance, especially in the chemical and petrochemical industries. All of these ideas lead very naturally to the discipline (and practice) of Closed Loop Real-Time Optimization.

CLRTO involves using large volumes of information from the process being optimized, as well as economic and environmental data affecting the operation of the process, to calculate optimum operating points for the process. All of this data used has associated uncertainty, be it the fluctuation in crude oil prices or the unpredictability of weather systems in the area. This uncertainty can cause the CLRTO system to generate solutions that are sub-optimal or infeasible with respect to the physical process being optimized. Conventional RTO systems either ignore this uncertainty altogether, or try to compensate for it in an indirect manner.

Stochastic Programming involves using information about the process model's uncertainty in formulating the optimization problem, such that the solution minimizes (in some sense) the undesirable effects of this uncertainty. In this thesis, ideas from a branch of Stochastic Programming, Probabilistic Programming, were tested on the gasoline blending optimization problem. The general gasoline blending optimization problem involves maximizing the profit obtained by selling the blend, subject to blend quality specifications, feedstock availabilities and product (blend) demands. All of the economic factors and constraint parameters that are used in the formulation of this optimization problem are uncertain to some degree. The focus of this work was using Probabilistic Programming techniques to address the parametric uncertainty associated with the blend quality specification constraints.

A Feasibility Horizon RTO formulation, derived from the THRTO approach presented by Singh [1997], was developed in this work. This formulation offers flexibility to specify a feasibility horizon for the optimization problem, instead of using the fixed horizon (given by the end of the blend) used in the earlier work. Feasibility Horizon RTO, with different feasibility horizons, was applied on a case study problem in two forms; 1) FHRTO with deterministic blend quality constraints, and 2) FHRTO with probabilistic constraints. The results from these two forms of the FHRTO formulation were then compared to the results obtained by applying the full THRTO approach, as well as conventional blending optimization algorithms.

Results from the gasoline blending case study confirm that conventional *LP + bias* algorithms do not handle uncertainty in the constraints very well. The formulations with chance constraints perform significantly better in the case study than those with deterministic constraints (expected values used for uncertain parameters). Also, the FHRTO formulations (both deterministic as well as probabilistic) with longer feasibility horizons have a tendency to drive the blend being produced, into situations from where the optimizer cannot recover. A longer feasibility horizon requires predictions further into the future (which leads to higher uncertainty). Thus, although a longer optimization horizon allows the optimizer more flexibility (more variables) to aim for higher profit, the greater uncertainty in the constraints can lead to these infeasible blend situations. For the case study used, the best performance was obtained by using the zero-step Feasibility Horizon RTO formulation with chance constraints (Chance Constrained Feasible over Current Interval RTO).

All probabilistic constraints used in this work were individual probabilistic constraints *i.e.*, each constraint was given a minimum probability of feasibility. Although a joint probabilistic constraint where the entire set of inequality constraints is given a minimum probability of feasibility is a more natural formulation, further work is needed to develop solution techniques needed to solve that problem. Further research is also needed to explore the role and effectiveness of the other Stochastic Programming techniques (besides Probabilistic Programming) in dealing with the uncertain constraints problem.

Two of the assumptions used while designing the gasoline blending optimization case study, were that, 1) the structure of the (Octane Number and *RVP*) blending models is known exactly, and 2) the time series disturbance model is known exactly. As discussed in Singh [1997], it can be very difficult to determine the true structure of most gasoline property blending models (especially Octane Number models). In practice, the disturbance models would be obtained by using Time Series Analysis [Box and Jenkins, 1976] and thus would also be uncertain to some extent. The effects of these uncertainties on the overall optimization problem were not addressed in this work, and need to be examined.

Another outstanding issue that needs further research is uncertainty in the optimization objective function. Objective function uncertainty, as opposed to constraint function uncertainty, does not (by itself) lead the optimizer to infeasible solutions. It can, however, lead the optimizer to solutions that are sub-optimal with respect to the actual process.

Bibliography

Anonymous (1997). Advanced control and information systems '97. *Hydrocarbon Processing* pp. 98–104.

Avriel, M. (1976). *Nonlinear Programming: Analysis and Methods*. Prentice-Hall. New Jersey.

Bain, M.L., K.W. Mansfield, J.G. Maphet, R.W. Szoke, W.H. Bosler and J.P. Kennedy (1993). Gasoline blending and integrated on-line optimization, scheduling and control system. *NPRA Comp. Conf.* Nov., paper no. CC–93–191, 129–139.

Biegler, L.T., L.E. Grossman and A.W. Westerberg (1985). A note on approximation techniques used for process optimization. *Comp. & Chem. Eng.* **9**(2), 201–206.

Box, G.E.P. and G.M. Jenkins (1976). *Time Series Analysis: Forecasting and Control*. Holden-Day. Oakland, CA.

Cao, S. and R.R. Rhinehart (1995). An efficient method for on-line identification of steady state. *J. Proc. Control* 5(6), 363–374.

Charnes, A. and W.W. Cooper (1963). Deterministic equivalents for optimizing and satisficing under chance constraints. *Operations Research* **11**, 18–39.

Coleman, T., M.A. Branch and A. Grace (1999). *Optimization Toolbox User's Guide*. The Mathworks, Inc.

Constantinou, L., K. Bagherpour, R. Gani, J.A. Klein and D.T. Wu (1996). Computer aided product design: Problem formulations, methodology and applications. *Comp. & Chem. Eng.* **20**, 685–702.

Crowe, C.M. (1988). Recursive identification of gross errors in linear data reconciliation. *AIChE J.* **34**, 541.

Crowe, C.M. (1996). Data reconciliation - progress and challenges. *J. Proc. Control* **6**, 89–98.

Dantzig, G.B. (1955). Linear programming under uncertainty. *Management Science* **1**, 197–206.

De Hennin, S.R. (1994). Structural Decisions in On-line Process Optimization. PhD thesis. University of London. U.K.

DeWitt, C.W., L.S. Lasdon, A.D. Waren, D.A. Brenner and S.A. Melhem (1989). Omega: An improved gasoline blending system for texaco. *Interfaces* **19**, 85–101.

Edgar, T.F. and D.M. Himmelblau (1988). *Optimization of Chemical Processes.* McGraw-Hill. New York.

Fletcher, R. (1987). *Practical Methods of Optimization.* John Wiley & Sons. New York.

Forbes, J.F. (1994). Model Structure and Adjustable Parameter Selection for Operations Optimization. PhD thesis. McMaster University. Canada.

Forbes, J.F. and T.E. Marlin (1993). Model updating considerations for real-time optimization. IFAC Workshop PCPI '93. Dusseldorf, Germany.

Forbes, J.F. and T.E. Marlin (1994). Model accuracy for economic optimizing controllers: The bias update case. *Ind. Eng. Chem. Res.* **33**, 1919–1929.

Forbes, J.F. and T.E. Marlin (1996). Design cost: A systematic approach to technology selection for model based real-time optimization systems. *Comp. & Chem. Eng.* **20**, 717–734.

Forbes, J.F., T.E. Marlin and J.F. MacGregor (1994). Model adequacy requirements for optimizing plant operations. *Comp. & Chem. Eng.* **18**, 497–510.

Friedman, Y. and G.V. Reklaitis (1975). Flexible solutions to linear programs under uncertainty: inequality constraints. *AIChE J.* **21**(1), 77–83.

Garcia, C.E. and M. Morari (1981). Optimal operation of integrated processing systems. *AIChE J.* **27**, 960.

Garcia, C.E., D.M. Prett and M. Morari (1989). Model predictive control: Theory and practice - a survey. *Automatica* **25**, 335–348.

Gary, J.H. and G.E. Handwerk (1994). *Petroleum Refining Technology and Economics.* 3rd ed.. Marcel Dekker. New York.

Gill, P.E., W. Murray and M.H. Wright (1981). *Practical Optimization.* Academic Press.

Givens, L. (1985). Modern petroleum refining: An overview. *Automotive Engineering* **93**, 69–77.

Grosdidier, Pierre (1997). Economics of blend giveaway. *Hydrocarbon Processing* pp. 55–60.

Healy, W.C. Jr., C.W. Maassen and R.T. Peterson (1959). A new approach to blending octanes. In: *Proc. 24th Midyear Meeting of the American Petroleum Institute's Division of Refining.* Vol. 39. New York.

Honeywell Hi-Spec Solutions (viewed 24 September, 2000). Profit.plus: Blend optimization. *http://www.hispec.com/docs/display.asp?doctype=1&doc=54.*

Hyprotech Ltd. (viewed 24 September, 2000). Hysys.rto+. *http://www.hyprotech.com/hysys/rto/rto.htm.*

Kelly, J.D. (1999). Reconciliation of process data using other projection matrices. *Comp. & Chem. Eng.* **23**(6), 785–789.

Krishnan, S., G.W. Barton and J.D. Perkins (1992). Robust parameter estimation in on-line optimization- part I: Methodology and simulated case-study. *Comp. & Chem. Eng.* **16**(6), 545–562.

Lobo, M.S., L. Vandenberghe, S. Boyd and H. Lebret (n.d.). Applications of second-order cone programming.

Loeblein, C. (1997). Analysis and Structural Design of On-line Process Optimization Systems. PhD thesis. Imperial College of Science, Technology and Medicine. U.K.

Maranas, C.D. (1997). Optimization accounting for property prediction uncertainty in polymer design. *Comp. & Chem. Eng.* **21**(Suppl.), 1019–1024.

Mehrotra, S. (1992). On the implementation of a primal-dual interior point method. *SIAM J. Opt.* **2**, 575–601.

Miletic, I. and T.E. Marlin (1998). Online statistical results analysis in real time operations optimization. *Ind. Eng. Chem. Res.* **37**, 3670–3684.

Miller, L.B. and H. Wagner (1965). Chance-constrained programming with joint constraints. *Operations Research* **13**, 930–945.

Morris, W.E., W.E. Smith and R.D. Snee (1994). Interaction blending equations enhance reformulated gasoline profitability. *Oil & Gas J.* **Jan. 17**, 54–58.

Palmer, F.H. and A.M. Smith (1985). The performance and specification of gasolines. In: *Technology of Gasoline* (E.G. Hancock, Ed.). Blackwell Scientific Publications. London.

Prekopa, Andras (1973). Contributions to the theory of stochastic programming. *Mathematical Programming* **4**, 202–221.

Prekopa, Andras (1995). *Stochastic Programming.* Kluwer Academic. Dordrecht, Netherlands.

Ramsey, J.R. Jr. and P.B. Truesdale (1990). Blend optimization integrated into refinery-wide strategy. *Oil & Gas J.* **Mar. 19**, 40–44.

Rigby, A., and L.S. Lasdon and A.D. Warren (1995). The evolution of texaco's blending systems: From omega to starblend. *Interfaces* **25**, 64–83.

Roush, W.B., R.H. Stock, T.L. Craverer and T.H. D'Alfonso (1994). Using chance-constrained programming for animal feed formulation at agway. *Interfaces* **24**, 53–58.

Rusin, M.H., H.S. Chung and J.F. Marshall (1981). A "transformation" method for calculating the research and motor octane numbers of gasoline blends. *Ind. Eng. Chem. Fundam.* **20**, 195–204.

Schwarm, A. and M. Nikolaou (1999). Chance constrained model predictive control. *AIChE J.* **45**(8), 1743–1752.

Singh, A. (1997). Modeling and model updating in the real-time optimization of gasoline blending. Master's thesis. University of Toronto.

Stewart, W.E. (1959*a*). Predict octanes for gasoline blends. *Petroleum Refiner* **38**, 135–139.

Stewart, W.E. (1959*b*). Predict rvp of blends accurately. *Petroleum Refiner* **38**, 231–234.

Sullivan, T.L. (1990). Refinery-wide blend control and optimization. *Hydrocarbon Processing* May, 93–96.

Terwiesch, P., D. Ravemark, B. Schenker and D.W.T. Rippin (1998). Semi-batch process optimization under uncertainty: Theory and experiments. *Comp. & Chem. Eng.* **22**(1-2), 201–213.

The Mathworks, Inc. (1999*a*). Matlab version 5.3.

The Mathworks, Inc. (1999*b*). Optimization toolbox 2.0.

Tjoa, I.B. and L.T. Biegler (1990). Simultaneous strategies for data reconciliation and gross error detection of nonlinear systems. *AIChE J. Annual Meeting.*

Tong, H. and C.M. Crowe (1997). Detecting persistent gross errors by sequential analysis of principal components. *AIChE J.* **43**, 1242–1249.

Treiber, S., R.S. McLeod, Y. Faitakis and R.L. Hutchings (1998). Rfg: the challenge to conventional blending technology. *Hydrocarbon Processing* pp. 101–104.

Vajda, S. (1972). *Probilistic programming.* Academic Press. New York.

van Krevelen, D.W. (1990). *Properties of Polymers: their correlation with chemical structure; their numerical estimation and prediction from additive group contributions.* 3rd ed.. Elsevier. Amsterdam.

Vazquez-Esparragoza, J.J., G.A. Iglesias-Silva, M.W. Hlavinka and J. Bullin (1992). How to estimate RVP of blends. *Hydrocarbon Processing* **Aug.**, 135–138.

White, J. and F. Hall (1992). Gasoline blending optimization cuts use of expensive components. *Oil & Gas J.* **Nov. 9**, 81–84.

Wright, S.J. (1997). *Primal-Dual Interior-Point Methods.* SIAM. Philadelphia.

Zhang, Y. and J.F. Forbes (2000). Extended design cost: A performance criterion for real-time optimization systems. *Comp. & Chem. Eng.* **24**(8), 1829–1841.

Zhang, Y., D. Nadler and J.F. Forbes (in press 2000). Results analysis for trust constrained real-time optimization. *J. Proc. Control.*

Appendix A

List of Symbols

$a_1, ..., a_6$	Ethyl RT-70 Octane Number model correlation coefficients in Equations 3.3
A	aromatic content (% by volume) in Equations 3.3
$\mathbf{A}$	coefficient matrix for linear inequality constraints
$\mathbf{b}$	right hand side vector for linear inequality constraints
$\mathbf{c}$	objective function cost parameters
$\mathbf{d}$	vector of maximum expected violations in Problem 2.9
$\mathbf{e}$	vector of economic data for the blending optimization problem
$f()$	objective function
F	normal probability distribution function
$\mathbf{g}$	vector of nonlinear functions in the blend quality specification constraints
$\mathbf{h}$	vector of inequality constraints
$\mathbf{H}$	coefficient matrix for the demand and availability constraints
k	current RTO interval
$\mathbf{m}$	vector of feedstock *MON*
n_{comp}	number of components or feedstock
n_{qual}	number of quality or property constraints
$\mathbf{o}$	vector of feedstock olefin content (% by volume)
p	selling price of a blended gasoline grade
$P()$	probability of an event
$\mathbf{Q}$	matrix of feedstock qualities
$\mathbf{r}$	vector of feedstock *RON*
$\mathbf{s}$	blend quality specifications vector
t	RTO time interval
u	white noise time series (source for quality disturbances)
$\mathbf{v}$	vecor of feedstock *RVP* blending indices
w	measurement noise time series
$\mathbf{x}$	decision variables; vector of feedstock flowrates
y	feedstock quality disturbance time series

z^{-1}	backward shift operator
$\boldsymbol{\alpha}$	matrix of maximum variation in inequality constraint parameters in Problem 2.13
$\boldsymbol{\beta}$	bias vector in LP + bias updating and NLP + bias updating problems
$\boldsymbol{\delta}$	vector of back-offs from active constraints in Problem 3.9
$\boldsymbol{\mu}_j$	vector orthogonal to $\boldsymbol{\alpha}_j$ (Equation 2.14)
$\boldsymbol{\xi}$	vector of constraint parameters in Problem 2.1
π_j	minimum probability of feasibility for a probabilistic constraint, j
$\boldsymbol{\rho}$	vector of blend demands and feedstock availabilities
σ	standard deviation
τ	normal distribution constant defined by Equation 3.8
$\boldsymbol{\chi}$	matrix $\boldsymbol{\alpha}$, modified for constraints with column-wise linear dependence (see Problem 2.15)
ω_i	volume fraction of component (feedstock), i
$\boldsymbol{\ell}$	right hand side vector for linearized quality constraints (Equation 3.19)

Appendix B

Interpretation of CCP

The i^{th} blend quality constraint in Problem 3.6 can be expanded to give:

$$(\widetilde{q}_{i,reformate} - s_i)x_{reformate} + (\widetilde{q}_{i,LSR} - s_i)x_{LSR} + (\widetilde{q}_{i,catgas} - s_i)x_{catgas} + \dots$$
$$+ (\widetilde{q}_{i,butane} - s_i)x_{butane} + (\widetilde{q}_{i,alky} - s_i)x_{alky} \leq 0 \quad \text{(B.1)}$$

(in the form $g_i(\mathbf{x}) \leq 0$)
the variance of which is:

$$E(g^2(\mathbf{x})) = E[(\widetilde{q}_{i,reformate} - s_i)^2 x^2_{reformate}] + E[(\widetilde{q}_{i,LSR} - s_i)^2 x^2_{LSR}] + \dots \quad \text{(B.2)}$$

for known $\mathbf{x}$:

$$E(g^2(\mathbf{x})) = \sum_j \sigma_j^2 x_j^2 \quad \text{(B.3)}$$

where j is the feedstock index and σ_j is the standard deviation of the i^{th} quality index of the j^{th} feedstock (given).

Appendix C

Bounding joint probabilities

For n events, A_i, Boole's inequality states that:

$$P(A_1 \cup A_2 \cup A_3 \cup ...A_n) \leq \sum_{i=1}^{n} P(A_i) \tag{C.1}$$

taking the complement of both sides and using the identity:
if $P(A) \leq P(B)$, then $P(\bar{A}) \geq P(\bar{B})$, we get:

$$1 - P(A_1 \cup A_2 \cup A_3 \cup ...A_n) \geq 1 - \sum_{i=1}^{n} P(A_i) \tag{C.2}$$

or

$$P(\bar{A}_1 \cap \bar{A}_2 \cap \bar{A}_3 \cap ...\bar{A}_n) \geq 1 - \sum_{i=1}^{n} P(A_i) \tag{C.3}$$

Inverting the above:

$$P(A_1 \cap A_2 \cap A_3 \cap ...A_n) \geq 1 - \sum_{i=1}^{n} P(\bar{A}_i) \tag{C.4}$$

Let the 3 probabilistic constraints be:

$$P(A_1) \equiv P(f_1(q_1, \mathbf{x}) \leq 0) \geqslant \alpha_1 \tag{C.5}$$
$$P(A_2) \equiv P(f_2(q_2, \mathbf{x}) \leq 0) \geqslant \alpha_2 \tag{C.6}$$
$$P(A_3) \equiv P(f_3(q_3, \mathbf{x}) \leq 0) \geqslant \alpha_3 \tag{C.7}$$

i.e., A_i is the event that the i^{th} constraint is satisfied.
Therefore, for 3 chance constraints with a 95% probability of feasibility each, the lower limit on the joint probability level is given by:

$$P(A_1 \cap A_2 \cap A_3) \geq 1 - 3 * 0.05 = 0.85 = 85\% \tag{C.8}$$

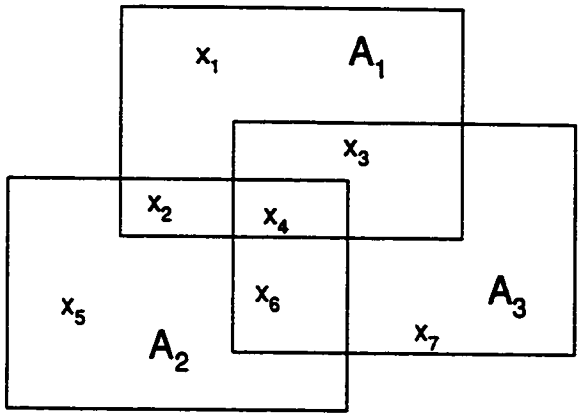

Figure C.1: Bounding joint probabilities

The above inequality gives us the lower bound on infeasibility of the joint probabilistic constraints. Intuitively, it means that there is a chance that all the constraints could be violated 5% of the time and that all violations could be independent (only one constraint violated in any one event).

Thus if we want at least a 95% level on the joint probability level, the individual chance constraints should be at the level given by:

$$0.95 = 1 - 3 * (1 - \alpha) \tag{C.9}$$

$$\alpha = 0.9833 = 98\frac{1}{3}\% \tag{C.10}$$

The example above is illustrated in Figure C.1, where the regions of feasibility of each constraint are separated into seven regions: $x_1, x_2, ..., x_7$. The problem of bounding the joint probability that all constraints are satisfied can now be formulated.

C.1 Lower bound

The lower bound on the probability that all three constraints are satisfied is given by the solution to the following problem.

$$\begin{aligned}
&\min_{x} x_4 && \text{(C.11)}\\
&s.t.\\
&x_1 + x_2 + x_3 + x_4 \geqslant 0.95\\
&x_2 + x_4 + x_5 + x_6 \geqslant 0.95\\
&x_3 + x_4 + x_6 + x_7 \geqslant 0.95\\
&x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 \leqslant 1\\
&0 \leqslant x_i \leqslant 1
\end{aligned}$$

The solution to the lower bound problem is, $x_4 = 0.85$. Thus, if there are three individual probabilistic constraints with a 95% probability of feasibility, the lowest joint probability of feasibility is 85%.

C.2 Upper bound

The highest joint probability of feasibility is given by the solution to:

$$\max_{\mathbf{x}} x_4 \tag{C.12}$$
$$s.t.$$
$$x_1 + x_2 + x_3 + x_4 \geqslant 0.95$$
$$x_2 + x_4 + x_5 + x_6 \geqslant 0.95$$
$$x_3 + x_4 + x_6 + x_7 \geqslant 0.95$$
$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 \leqslant 1$$
$$0 \leqslant x_i \leqslant 1$$

The solution to the upper bound problem is, $x_4 = 1$.

The theorem for n simultaneous, independent events cannot be used here because, the theorem assumes the individual constraints to be of the form: $P(A_i) = \alpha_i$ (equality constraints). Probabilistic constraints are inequalities given by $P(A_i) \geq \alpha_i$. The theorem for n independent events occurring simultaneously states that the probability that at least one constraint will be violated is given by:

$$P(\bar{A}_1 \cup \bar{A}_2 \cup \bar{A}_3) = P(\bar{A}_1) + P(\bar{A}_2) + P(\bar{A}_3) - P(\bar{A}_1 \cup \bar{A}_2) - ...$$
$$P(\bar{A}_2 \cup \bar{A}_3) - P(\bar{A}_3 \cup \bar{A}_1) + 2P(\bar{A}_1 \cap \bar{A}_2 \cap \bar{A}_3) \tag{C.13}$$

for independent events A_1, A_2, A_3.

for $P(A_1) = P(A_2) = P(A_3) = 0.95$:

$$P(\bar{A}_1 \cup \bar{A}_2 \cup \bar{A}_3) = 3 \times 0.05 - 3 \times 0.05^2 + 2 \times 0.05^3 = 0.142625 = 14.26\% \tag{C.14}$$

or

$$P(A_1 \cap A_2 \cap A_3) = 1 - 0.1426 = 85.74\% \tag{C.15}$$

using the reverse argument (again for equality probabilities), for $P(A_1 \cap A_2 \cap A_3) = 0.95$, $P(A_1) = (0.95)^{1/3} = 0.98305 = 98.305\%$

Appendix D

Linearization of the Ethyl model

D.1 The *RON* constraint

$$\nabla_{\mathbf{x}} g_{RON} = \mathbf{r}^T + a_1\{\mathbf{r}^T diag(\mathbf{r}-\mathbf{m}) - (\mathbf{1}^T\mathbf{x})^{-1}[\mathbf{r}^T\mathbf{x}(\mathbf{r}-\mathbf{m})^T + (\mathbf{r}-\mathbf{m})^T\mathbf{x}\mathbf{r}^T]...$$
$$... + (\mathbf{1}^T\mathbf{x})^{-2}(\mathbf{r}-\mathbf{m})^T\mathbf{x}\mathbf{r}^T\mathbf{x}\mathbf{1}^T\} + a_2\{\mathbf{o}_{sq}^T - 2(\mathbf{1}^T\mathbf{x})^{-1}\mathbf{o}^T\mathbf{x}\mathbf{o}^T + (\mathbf{1}^T\mathbf{x})^{-2}\mathbf{o}^T\mathbf{x}\mathbf{o}^T\mathbf{x}\mathbf{1}^T\}...$$
$$... + a_3\{\mathbf{b}_{sq}^T - 2(\mathbf{1}^T\mathbf{x})^{-1}\mathbf{b}^T\mathbf{x}\mathbf{b}^T + (\mathbf{1}^T\mathbf{x})^{-2}\mathbf{b}^T\mathbf{x}\mathbf{b}^T\mathbf{x}\mathbf{1}^T\} \quad \text{(D.1)}$$

$$\nabla_{\mathbf{r}} g_{RON} = \mathbf{x}^T + a_1\{\mathbf{x}^T \mathrm{diag}(2\mathbf{r}-\mathbf{m}) - (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{x}^T(2\mathbf{r}-\mathbf{m})\mathbf{x}^T\} \quad \text{(D.2)}$$

$$\nabla_{\mathbf{m}} g_{RON} = a_1\{-\mathbf{x}^T \mathrm{diag}(\mathbf{r}) + (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{x}^T\mathbf{r}\mathbf{x}^T\} \quad \text{(D.3)}$$

D.2 The *MON* constraint

$$\nabla_{\mathbf{x}} g_{MON} = \mathbf{m}^T + a_4\{\mathbf{m}^T diag(\mathbf{r}-\mathbf{m}) - (\mathbf{1}^T\mathbf{x})^{-1}[\mathbf{m}^T\mathbf{x}(\mathbf{r}-\mathbf{m})^T + (\mathbf{r}-\mathbf{m})^T\mathbf{x}\mathbf{m}^T]...$$
$$... + (\mathbf{1}^T\mathbf{x})^{-2}(\mathbf{r}-\mathbf{m})^T\mathbf{x}\mathbf{m}^T\mathbf{x}\mathbf{1}^T\} + a_5\{\mathbf{o}_{sq}^T - 2(\mathbf{1}^T\mathbf{x})^{-1}\mathbf{o}^T\mathbf{x}\mathbf{o}^T + (\mathbf{1}^T\mathbf{x})^{-2}\mathbf{o}^T\mathbf{x}\mathbf{o}^T\mathbf{x}\mathbf{1}^T\}...$$
$$...+2\tilde{a}_6(\mathbf{1}^T\mathbf{x})^{-1}[\mathbf{b}_{sq}^T\mathbf{x} - (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{b}^T\mathbf{x}\mathbf{b}^T\mathbf{x}]\times\{\mathbf{b}_{sq}^T - 2(\mathbf{1}^T\mathbf{x})^{-1}\mathbf{b}^T\mathbf{x}\mathbf{b}^T + (\mathbf{1}^T\mathbf{x})^{-2}\mathbf{b}^T\mathbf{x}\mathbf{b}^T\mathbf{x}\mathbf{1}^T\}...$$
$$... - \tilde{a}_6(\mathbf{1}^T\mathbf{x})^{-2}[\mathbf{b}_{sq}^T\mathbf{x} - (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{b}^T\mathbf{x}\mathbf{b}^T\mathbf{x}]^2\mathbf{1}^T \quad \text{(D.4)}$$

$$\nabla_{\mathbf{r}} g_{MON} = a_4\{\mathbf{x}^T \mathrm{diag}(\mathbf{m}) - (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{x}^T\mathbf{m}\mathbf{x}^T\} \quad \text{(D.5)}$$

$$\nabla_{\mathbf{m}} g_{MON} = \mathbf{x}^T + a_4\{\mathbf{x}^T \mathrm{diag}(\mathbf{r}-2\mathbf{m}) - (\mathbf{1}^T\mathbf{x})^{-1}\mathbf{x}^T(\mathbf{r}-2\mathbf{m})\mathbf{x}^T\} \quad \text{(D.6)}$$

Appendix E

Non-associativity of Ethyl RT-70 models

For the Ethyl RT-70 models, blending of a set of feedstocks is not associative. For an illustration, consider a scenario where three feedstocks, say a, b and, c are to be blended. The quantities of these three feedstocks are given by x_a, x_b, and x_c, while the octane number vectors are given by q_a , q_b, and q_c, where $q_i = \left[\begin{array}{cc} RON_i & MON_i \end{array} \right]^T$.

If the blending function $B(n, m, o) = q_{(n,m,o)}$ represents the qualities of the blend obtained by mixing feedstocks n, m, and o, the Ethyl RT-70 blending rules do not obey the following associative blending rule:

$$B(a, b, c) \neq B(d, c) \tag{E.1}$$

where $q_d = B(a, b)$ *i.e.*, d is the blend obtained by mixing feedstocks a and b.

This non-associative blending behaviour means that time horizon based blending optimization approaches will not work for the Ethyl RT-70 octane number models. Time horizon based blending optimization depends on predicting future feedstock qualities and optimizing the feedstock flows for current and future RTO intervals. The non-associative behaviour of the Ethyl models implies that, even if the predictions for current and future feed qualities were exact, the qualities predicted for the end of the blend can be different from the actual qualities obtained at the end of the batch.

Appendix F

Feedstock quality prediction error variance

The overall transfer function for the measured feedstock qualities as given by Equation 4.4:

$$\widehat{y}_k = y_k + w_k = \frac{0.39z^{-1}}{(1-0.61z^{-1})(1-z^{-1})}u_k + w_k \tag{F.1}$$

$$G(z^{-1}) = \frac{y_k}{u_k} = \frac{bz^{-1}}{1-cz^{-1}+az^{-2}} \tag{F.2}$$

where $a = 0.61$, $b = (1-0.61) = 0.39$, and $c = (1+0.61) = 1.61$.
From the overall transfer function:

$$y_k = cy_{k-1} - ay_{k-2} + bu_{k-1} \tag{F.3}$$
$$y_{k-1} = cy_{k-2} - ay_{k-3} + bu_{k-2} \tag{F.4}$$
$$y_{k-2} = cy_{k-3} - ay_{k-4} + bu_{k-3} \tag{F.5}$$

The variance of the measured feedstock qualities for the current interval is given by:

$$E\left[(\widehat{y}_k - y_k)^2\right] = E\left[(y_k + w_k - y_k)^2\right] = E[w_k^2] = \sigma_w^2$$

The variance for the one-step ahead prediction error of the measured feedstock qualities is given by:

$$\begin{aligned} E\left[(\widehat{y}_k - y_{k/k-1})^2\right] &= E\left[(cy_{k-1} - ay_{k-2} + bu_{k-1} + w_k - cy_{k-1} + ay_{k-2})^2\right] \\ &= E[(bu_{k-1} + w_k)^2] = b^2\sigma_u^2 + \sigma_w^2 \end{aligned}$$

Similarly, the two-step ahead through eleven-step prediction errors are given by the following expressions:

$$
\begin{aligned}
E\left[(\widehat{y}_k - y_{k/k-2})^2\right] &= E\left[(cy_{k-1} - ay_{k-2} + bu_{k-1} + w_k - cy_{k-1/k-2} + ay_{k-2})^2\right] \\
&= E[(c\{y_{k-1} - y_{k-1/k-2}\} + bu_{k-1} + w_k)^2] \\
&= E[(cbu_{k-2} + bu_{k-1} + w_k)^2] \\
&= b^2(1+c^2)\sigma_u^2 + \sigma_w^2
\end{aligned}
$$

$$
\begin{aligned}
E\left[(\widehat{y}_k - y_{k/k-3})^2\right] &= E\left[(cy_{k-1} - ay_{k-2} + bu_{k-1} + w_k - cy_{k-1/k-3} + ay_{k-2/k-3})^2\right] \\
&= E[(c\{y_{k-1} - y_{k-1/k-3}\} - a\{y_{k-2} - y_{k-2/k-3}\} + bu_{k-1} + w_k)^2] \\
&= E[(c^2bu_{k-3} - abu_{k-3} + cbu_{k-2} + bu_{k-1} + w_k)^2] \\
&= b^2[1 + c^2 + (c^2 - a)^2]\sigma_u^2 + \sigma_w^2
\end{aligned}
$$

$$
E\left[(\widehat{y}_k - y_{k/k-4})^2\right] = b^2[1 + c^2 + (c^2-a)^2 + c^2(c^2-2a)^2]\sigma_u^2 + \sigma_w^2
$$

$$
E\left[(\widehat{y}_k - y_{k/k-5})^2\right] = b^2[1 + c^2 + (c^2-a)^2 + c^2(c^2-2a)^2 + (c^4 - 3ac^2 + a^2)]\sigma_u^2 + \sigma_w^2
$$

$$
\begin{aligned}
E\left[(\widehat{y}_k - y_{k/k-6})^2\right] &= b^2[1 + c^2 + (c^2-a)^2 + c^2(c^2-2a)^2 + (c^4 - 3ac^2 + a^2)^2 \ldots \\
&\quad + c^2(c^4 - 4ac + 3a^2)^2]\sigma_u^2 + \sigma_w^2
\end{aligned}
$$

$$
\begin{aligned}
E\left[(\widehat{y}_k - y_{k/k-7})^2\right] &= b^2[1 + c^2 + (c^2-a)^2 + c^2(c^2-2a)^2 + (c^4 - 3ac^2 + a^2)^2 \ldots \\
&\quad + c^2(c^4 - 4ac + 3a^2)^2 + (c^6 - 5ac^4 + 6a^2c^2 - a^3)^2]\sigma_u^2 + \sigma_w^2
\end{aligned}
$$

$$
\begin{aligned}
E\left[(\widehat{y}_k - y_{k/k-8})^2\right] &= b^2[1 + c^2 + (c^2-a)^2 + c^2(c^2-2a)^2 + (c^4 - 3ac^2 + a^2)^2 \ldots \\
&\quad + c^2(c^4 - 4ac + 3a^2)^2 + (c^6 - 5ac^4 + 6a^2c^2 - a^3)^2 \ldots \\
&\quad + c^2(c^6 - 6ac^4 + 10a^2c^2 - 4a^3)^2]\sigma_u^2 + \sigma_w^2
\end{aligned}
$$

$$
\begin{aligned}
E\left[(\widehat{y}_k - y_{k/k-9})^2\right] &= b^2[1 + c^2 + (c^2-a)^2 + c^2(c^2-2a)^2 + (c^4 - 3ac^2 + a^2)^2 \ldots \\
&\quad + c^2(c^4 - 4ac + 3a^2)^2 + (c^6 - 5ac^4 + 6a^2c^2 - a^3)^2 \ldots \\
&\quad + c^2(c^6 - 6ac^4 + 10a^2c^2 - 4a^3)^2 \ldots \\
&\quad + (c^8 - 7ac^6 + 15a^2c^4 - 10a^3c^2 + a^4)^2]\sigma_u^2 + \sigma_w^2
\end{aligned}
$$

$$
\begin{aligned}
E\left[(\widehat{y}_k - y_{k/k-10})^2\right] &= b^2[1 + c^2 + (c^2-a)^2 + c^2(c^2-2a)^2 + (c^4 - 3ac^2 + a^2)^2 \ldots \\
&\quad + c^2(c^4 - 4ac + 3a^2)^2 + (c^6 - 5ac^4 + 6a^2c^2 - a^3)^2 \ldots \\
&\quad + c^2(c^6 - 6ac^4 + 10a^2c^2 - 4a^3)^2 + \ldots \\
&\quad + (c^8 - 7ac^6 + 15a^2c^4 - 10a^3c^2 + a^4)^2 + \ldots \\
&\quad + c^2(c^8 - 8ac^6 + 21a^2c^4 - 20a^3c^2 + 5a^4)^2]\sigma_u^2 + \sigma_w^2
\end{aligned}
$$

$$
\begin{aligned}
E\left[(\widehat{y}_k - y_{k/k-11})^2\right] &= b^2[1 + c^2 + (c^2 - a)^2 + c^2(c^2 - 2a)^2 + (c^4 - 3ac^2 + a^2)^2 ... \\
&+ c^2(c^4 - 4ac + 3a^2)^2 + (c^6 - 5ac^4 + 6a^2c^2 - a^3)^2 ... \\
&+ c^2(c^6 - 6ac^4 + 10a^2c^2 - 4a^3)^2 ... \\
&+ (c^8 - 7ac^6 + 15a^2c^4 - 10a^3c^2 + a^4)^2 ... \\
&+ c^2(c^8 - 8ac^6 + 21a^2c^4 - 20a^3c^2 + 5a^4)^2 ... \\
&+ (c^{10} - 9ac^8 + 28a^2c^6 - 35a^3c^4 + 15a^4c^2 - a^5)^2]\sigma_u^2 + \sigma_w^2
\end{aligned}
$$

Appendix G

Lagrange multipliers for THRTO - Ideal case

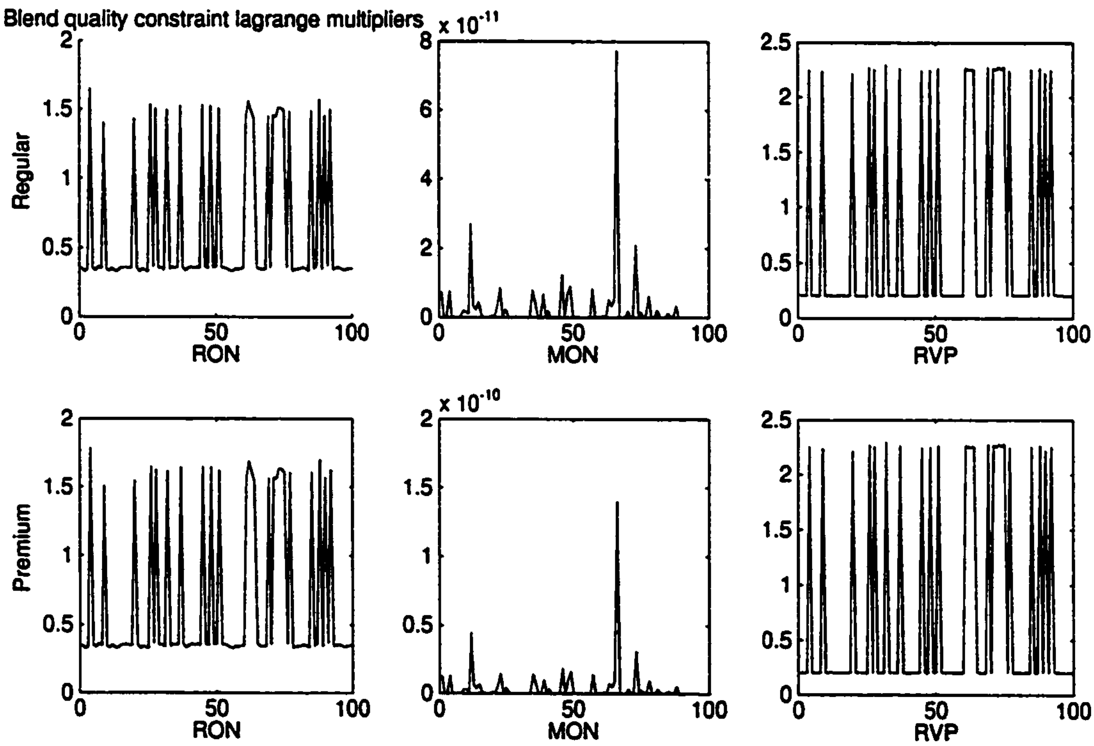

Figure G.1: Time Horizon RTO - Ideal case; Lagrange multipliers for blend quality constraints

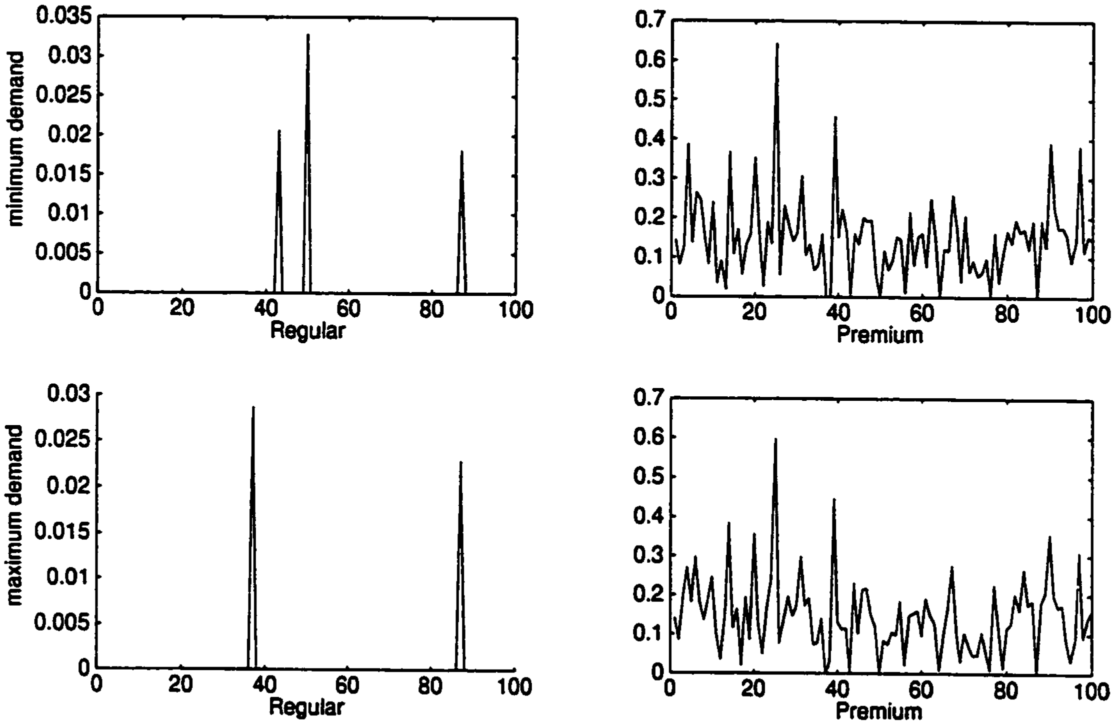

Figure G.2: Time Horizon RTO - Ideal case; Lagrange multipliers for blend demand constraints

Appendix H

Additional results for Feasibility Horizon RTO forms

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,507.88	10,990.78
Average daily profit ($/day)	29,882.23	71,426.52
Infeasible blends (days)	18	17

Table H.1: Results for 1-step ahead Feasibility Horizon RTO (deterministic constraints)

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,412.94	10,898.76
Average daily profit ($/day)	29,883.68	70,049.38
Infeasible blends (days)	24	24

Table H.2: Results for 2-step ahead Feasibility Horizon RTO (deterministic constraints)

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,318.10	10,796.64
Average daily profit ($/day)	29,776.68	69,179.88
Infeasible blends (days)	26	29

Table H.3: Results for 3-step ahead Feasibility Horizon RTO (deterministic constraints)

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,402.62	10,990.31
Average daily profit ($/day)	29,732.18	72,833.23
Infeasible blends (days)	3	3

Table H.4: Results for 1-step ahead FHRTO (chance constraints)

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,411.76	10,981.36
Average daily profit ($/day)	29,592.90	72,784.70
Infeasible blends (days)	5	3

Table H.5: Results for 2-step ahead FHRTO (chance constraints)

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,409.47	10,953.13
Average daily profit ($/day)	29,529.59	72,586.63
Infeasible blends (days)	6	4

Table H.6: Results for 3-step ahead FHRTO (chance constraints)

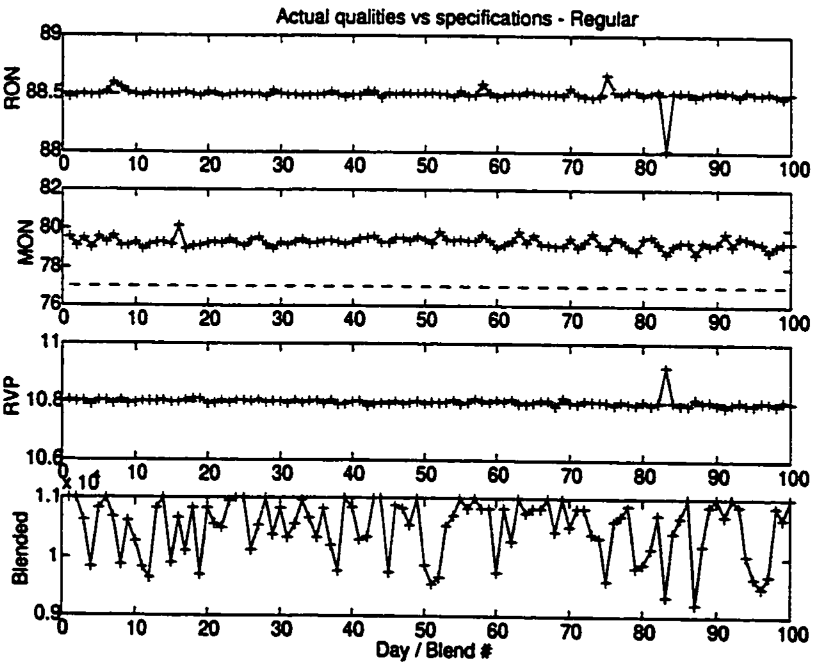

Figure H.1: Quality and amount blended profiles for 1-step ahead Feasibility Horizon RTO; regular

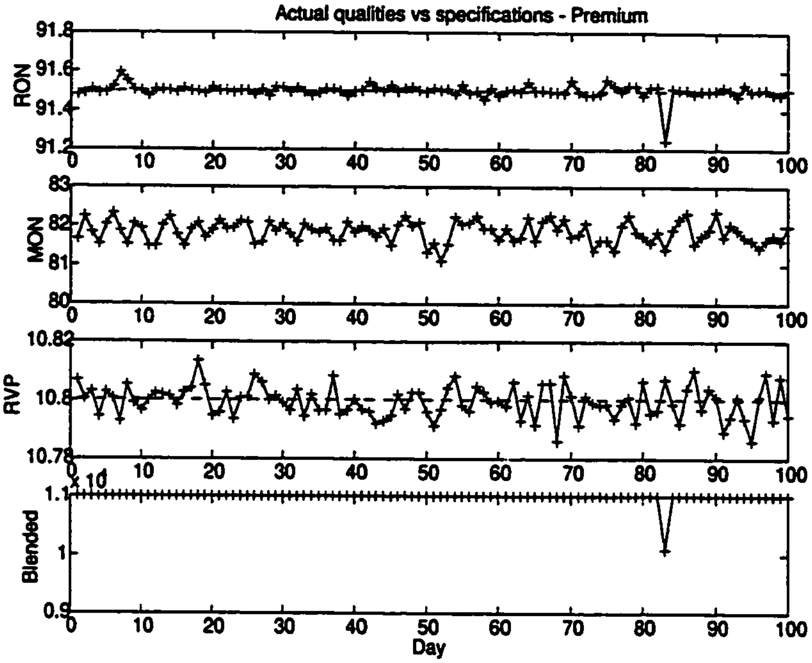

Figure H.2: Quality and amount blended profiles for 1-step ahead Feasibility Horizon RTO; premium

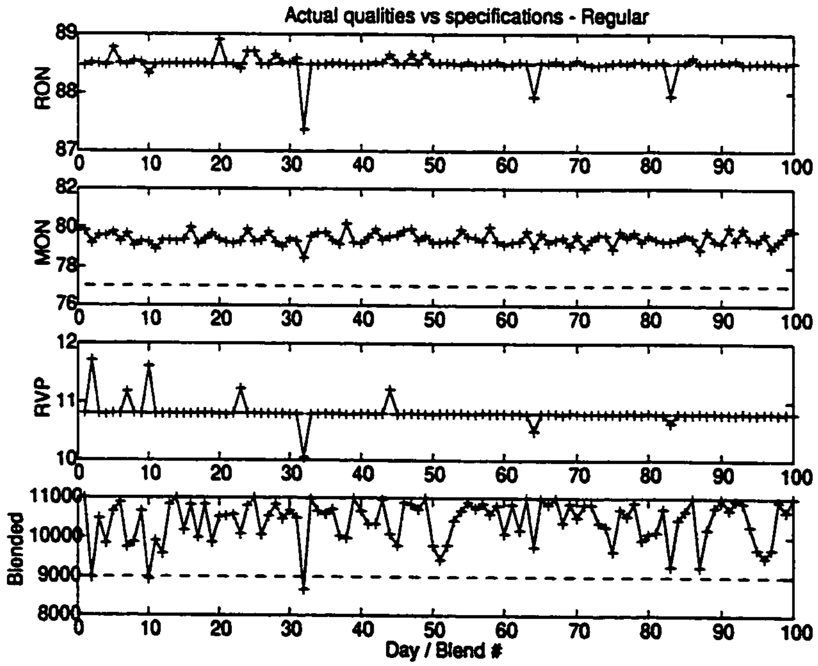

Figure H.3: Quality and amount blended profiles for 2-step ahead Feasibility Horizon RTO; regular

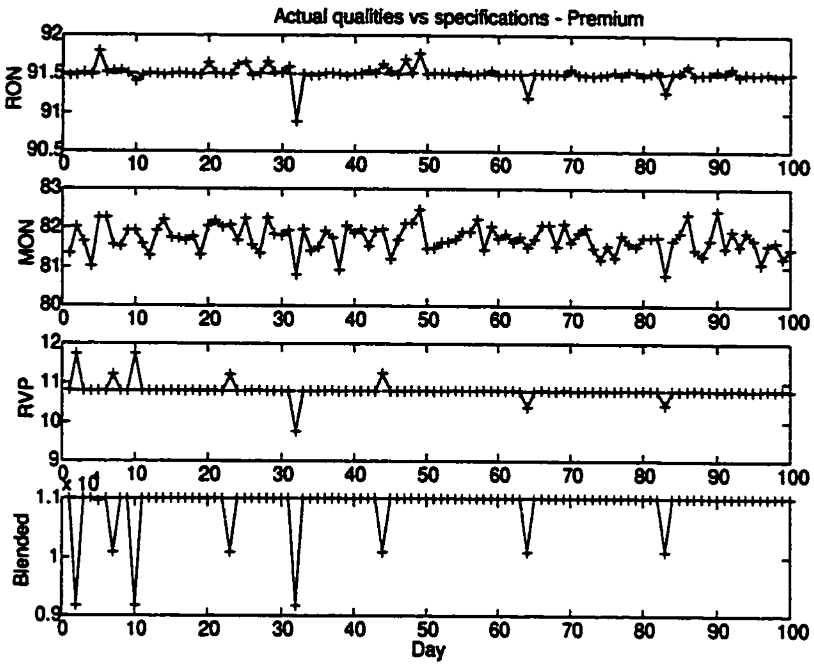

Figure H.4: Quality and amount blended profiles for 2-step ahead Feasibility Horizon RTO; premium

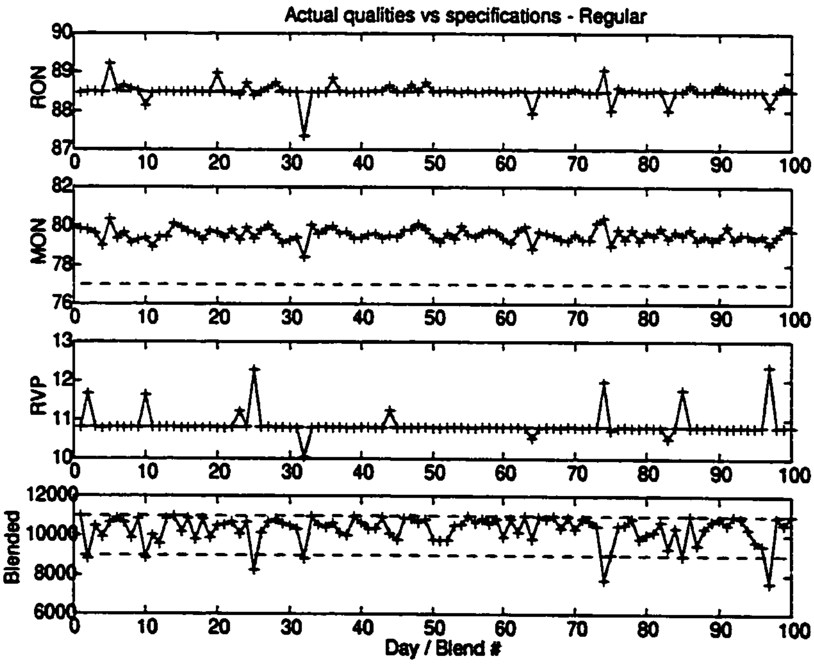

Figure H.5: Quality and amount blended profiles for 3-step ahead Feasibility Horizon RTO; regular

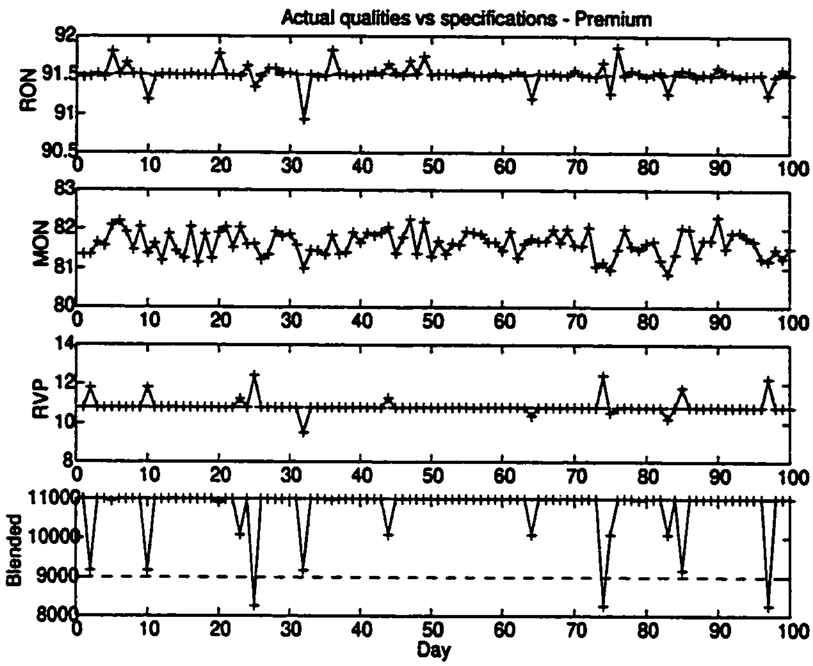

Figure H.6: Quality and amount blended profiles for 3-step ahead Feasibility Horizon RTO; premium

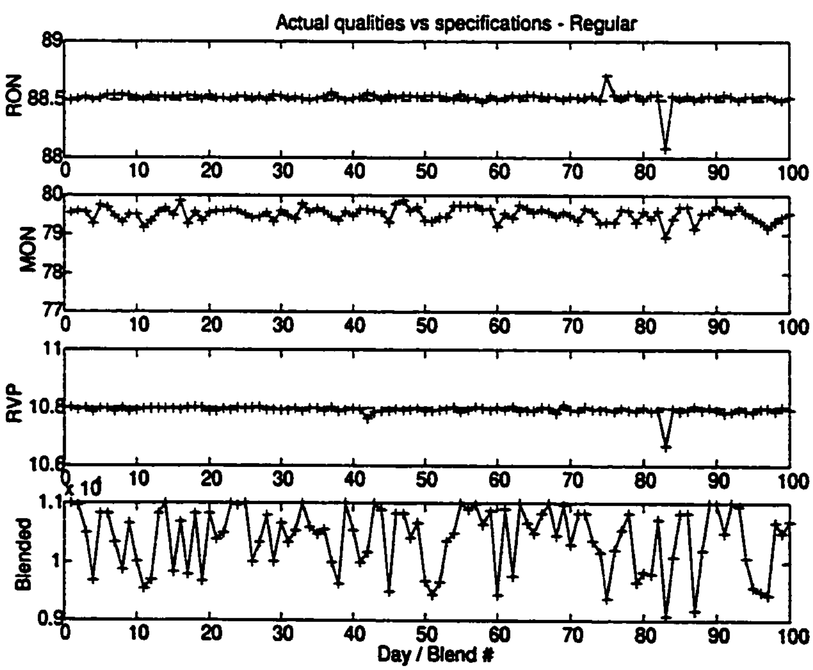

Figure H.7: Quality and amount blended profiles for 1-step ahead flexible Chance Constrained THRTO; regular

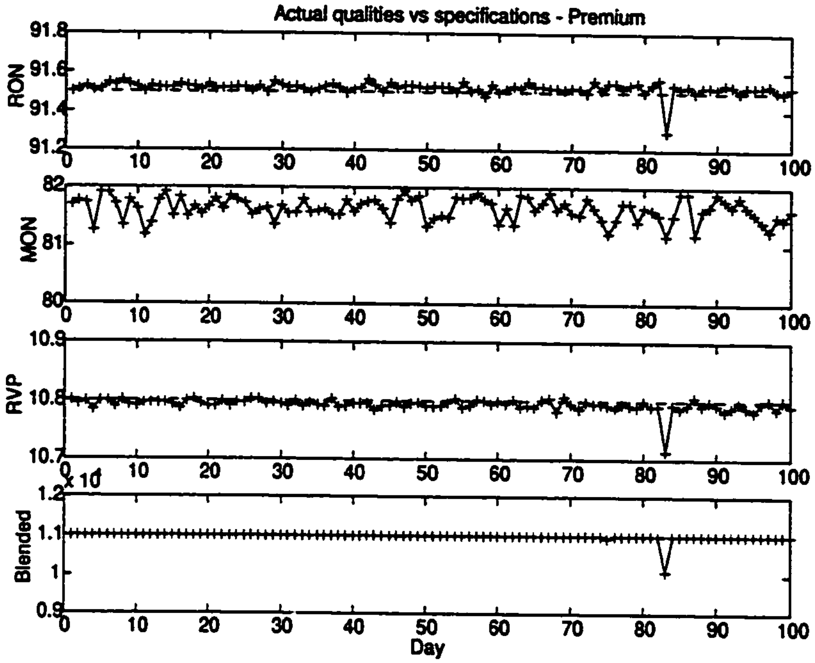

Figure H.8: Quality and amount blended profiles for 1-step ahead flexible Chance Constrained THRTO; premium

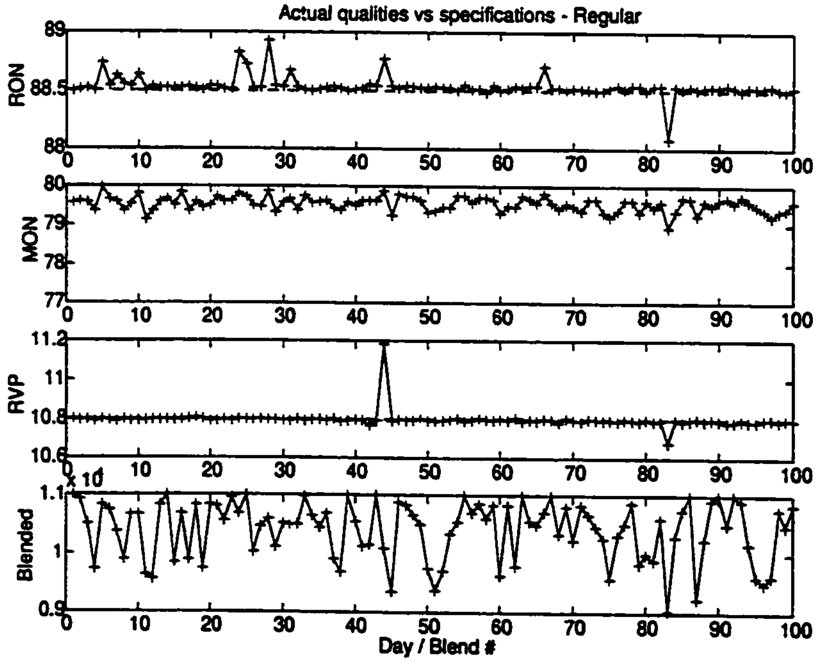

Figure H.9: Quality and amount blended profiles for 2-step ahead flexible Chance Constrained THRTO; regular

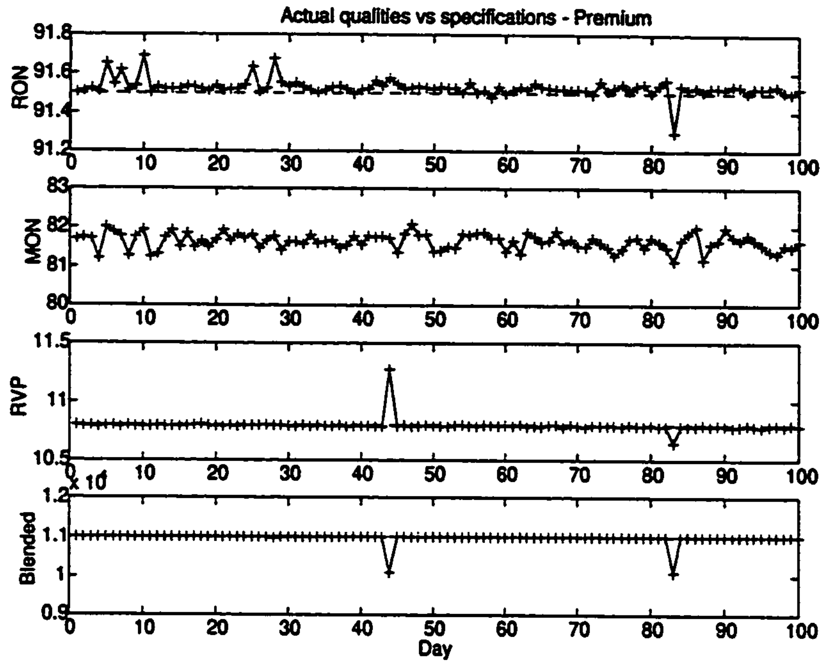

Figure H.10: Quality and amount blended profiles for 2-step ahead flexible Chance Constrained THRTO; premium

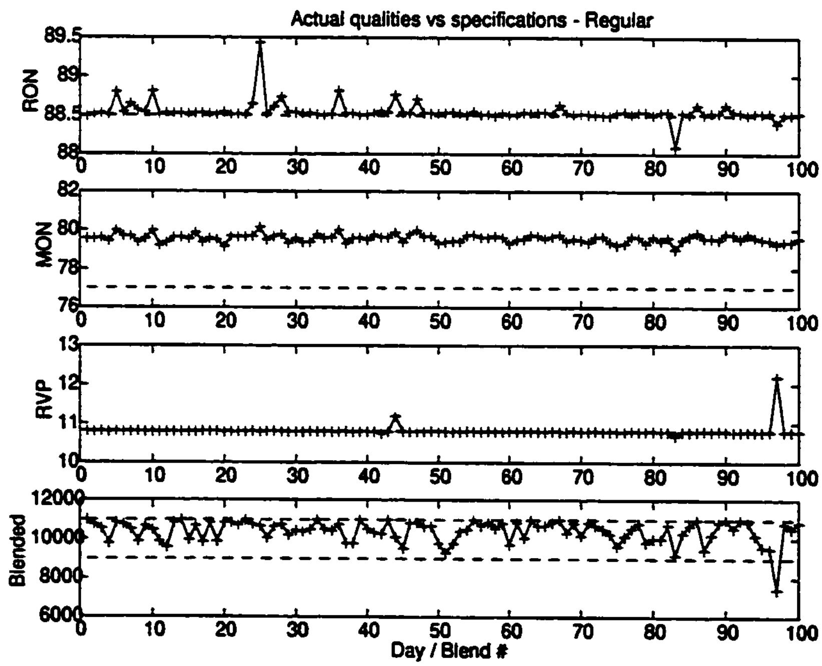

Figure H.11: Quality and amount blended profiles for 3-step ahead flexible Chance Constrained THRTO; regular

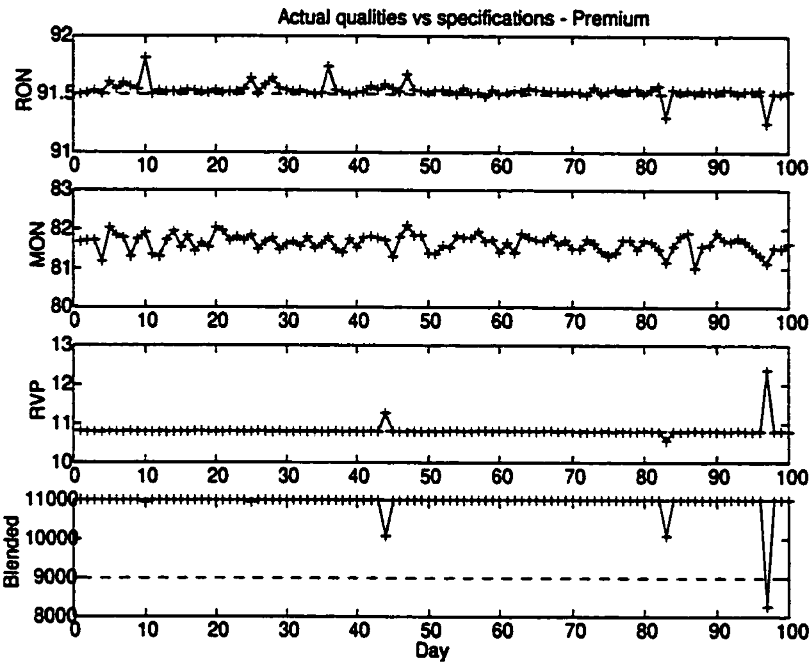

Figure H.12: Quality and amount blended profiles for 3-step ahead flexible Chance Constrained THRTO; premium

Appendix I

Demand over Blend Time-Horizon RTO Results

The results presented are those generated by using both demand over batch (day) as well as demand over sub-batch (each RTO interval) constraints. The demand over batch or day constraints are:

	Premium	*Regular*
Maximum demand, bbl/day	11000	9000
Minimum demand, bbl/day	11000	9000

Table I.1: Blend demand over day

The demand over sub-batch or RTO interval are:

	Premium	*Regular*
Maximum demand, bbl/interval	$22000/12 = 1833.33$	$4500/12 = 375$
Minimum demand, bbl/interval	$22000/12 = 1833.33$	$4500/12 = 375$

Table I.2: Blend demand over each interval

Thus, the maximum sub-batch demand limits are twice the maximum limits used in § 4.2 while the minimum sub-batch demand limits are half of those used in the case study in Chapter 4.

I.1 Ideal Case (all feedstock qualities known)

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,664.97	11,000.00
Average daily profit ($/day)	30,719.52	73,231.50

Table I.3: Results for the THRTO (Ideal case) algorithm

The average daily profit obtained by applying the THRTO algorithm with all feedstock qualities known exactly is, $ 103,951 per day.

I.2 Deterministic Constraints

The average overall daily profit obtained by using the Time-Horizon RTO formulation where both demand over batch (day) as well as demand over sub-batch (each RTO interval) are considered is \$97,688.

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,126.30	10,627.70
Average daily profit (\$/day)	29,061.97	68,626.39
Infeasible blends (days)	34	35

Table I.4: Results for the THRTO (deterministic constraints) algorithm

I.3 Probabilistic Constraints

The average overall daily profit obtained by using the Chance Constrained Time-Horizon RTO formulation where both demand over batch (day) as well as demand over sub-batch (each RTO interval) are considered is \$99,571.

Blend	*Regular*	*Premium*
Average daily production (bbl/day)	10,211.15	10,720.36
Average daily profit (\$/day)	28,993.46	70,577.21
Infeasible blends (days)	21	20

Table I.5: Results for the THRTO (chance constraints) algorithm