University of Alberta

Enhanced Design Process with CAD/CAE Integration and Smart Knowledge Capturing Method

by

Ganesh Prakash Gujarathi

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
Engineering Design

Department of Mechanical Engineering

Spring 2011

Edmonton, Alberta

Abstract

Generally, a design process involves multiple modeling and analysis interaction iterations accommodating changes and verifications. It is convenient to develop generative programs to automate some tedious and repetitive processes in order to minimize cycle times and the engineer's routine efforts associated with various design stages. The proposed process model incorporates embedded engineering knowledge and generative Computer Aided Design (CAD) and Computer Aided Engineering (CAE) analysis for partial automation of product development. For improved efficiency and ease of operation, the process used a CAD/CAE integration approach with smart program development mechanism for automated modeling and analysis.

A CAD/CAE integration method using a "Common Data Model" (CDM) containing all the required parametric information for both CAD modelling and CAE analysis is implemented. The CDM is used as a parametric data model repository and the supply source of input for those associative entities of CAD and CAE models and thus maintaining the associative dependencies among them. The pro-recorded journal file from the modeling and analysis software tool are used for rapid and easy program development of generative CAD and CAE.

Acknowledgements

I would like to express my deep and sincere gratitude to my supervisor, Dr. Yongsheng Ma. His wide knowledge and his logical way of thinking have been of great value for me. His understanding, encouraging and personal guidance have provided a good basis for the present thesis. I am also thankful to the Collaborative and Integrated Engineering Informatics Group for their technical support and feedback in my research.

My time at U of A was made enjoyable in large part due to the many friends and groups that became a part of my life. I am grateful for time spent with my roommates and friends. I am thankful to my family members in Edmonton for their hospitality and support which made me felt at home during my study abroad.

Lastly, I would like to thank my family for all their love and encouragement. For my parents, who raised me with a love of science and supported me in all my pursuits and for my sister Manjiri who always encouraged and guided me along the way.

Thank you.

Ganesh Prakash Gujarathi

Table of Contents

List of Tables

List of figures

List of Symbols

μ	- Viscosity, cp
ρl, ρg	- Density of liquid, gas, lb/ft3
A	- Distance of saddle from seam, in
AB	- Bolt circle area, sq.in
AR	- Base ring area, sq.in
As	- Area within skirt, sq.in
API°	- Gravity of oil
B	- Width of the saddle, in
BA	- Bolt area, sq.in
CB	- Bolt circle circumference, in
CD	- Drag coefficient of particle
CS	- Circumference of skirt, in
d	- Bolt circle diameter, in
dm	- Droplet diameter, micron
D	- Diameter of vessel, in or ft
Do	- Base plate outside diameter, in
Di	- Base plate inside diameter, in
Ds	- Outside skirt diameter, in
Dos	- Outside diameter of vessel, in
E	- Joint efficiency
h	- Liquid level in vessel, in
ht	- Skirt height, in
H	- Vessel total height, in
idl	- Inlet diverter length, in
ide	- Inlet diverter extension, in
idt	- Inlet diverter thickness, in
L	- Shell length, in
Leff	- Effective length, in

Lss	- Seam to seam length, in
MW	- Molecular weight
MEt	- mist extractor thickness, in
MESt	- Mist extractor support thickness, in
MESe	- Mist extractor support extension, in
N	- Number of bolts
P	- Operating pressure, psi
PC	- Maximum compression on base, lb/in2
Ql	- Liquid flow, bpd
Qg	- Gas capacity, MMscfd
R	- Outside radius of shell, in
RE	- Reynold's number
S	- Max. Stress, psi
Sb	- Max allowable stress in bolt, psi
SB	- Base plate maximum allowable stress, psi
Sh	- Stress of head material, psi
Ss	- Stress of shell material, psi
SG	- Specific gravity
SGg	- Specific gravity of gas
t	- Thickness, in
tb	- Base ring thickness, in
th	- Head thickness, in
tr	- Retention time of liquid, min
ts	- Shell thickness, in
tskirt	- Skirt thickness, in
T	- Temperature, °F.
Tb	- Tension in bolt, lb
vbl	- Vortex breaker length, in
vbe	- vortex breaker extension, in
vbt	- Vortex breaker thickness, in
vt	- Terminal velocity, ft/s

Wint	- Weight of internals, lb
W	- Operating weight, lb
Wh	- Head weight, lb
Ws	- Shell weight, lb
Ww	- Weight of water, lbs
Z	- Compressibility factor

List of Abbreviations

API	- Application Programming Interface
CAD	- Computer Aided Design
CAE	- Computer Aided Engineering
CDM	- Common Data Model
FEA	- Finite Element Analysis
GUI	- Graphical User Interface
KBE	- Knowledge Based Engineering

Chapter1

Introduction

Conventional design process for product development uses theoretical engineering calculation-based techniques. Such methodology usually involves multiple stages that are mostly performed interactively with the help of CAD tools and through modifications of design, i.e. a “trial and error” approach, trying to fit theoretical design methods to the specific problem.

Modern CAD modelling and CAE analysis cycles have become an inherent part of today’s product development process. In common practice, computer-aided design (CAD) and computer aided engineering analysis (CAE) processes are interwoven and iteratively carried out during the product development cycles. Computer-aided tools are also broadly used for process planning, tool design, production planning, automated machining, inspection, assembly, costing and collaboration development [1]. There are various commercially available CAD software tools such as Pro-E™, Solid works™, Siemens NX™, etc. Along with geometry development, another task involved with the design process is analysis and verification. Conventional analytical processes are being complemented by the modern computational methods, such as Finite Element Analysis (FEA). FEA breaks down a complex body of problem into discrete mesh elements, and solves the problem using a progressive and numerical method. This technology has changed the engineering analysis landscape and enabled mass availability of numerical solutions for highly complex linear and non-linear equations. Some commercial CAE tools are Ansys™, LS Dyna™, NX Nastran™, etc. Therefore, CAD technology coupled with CAE offers an effective cyclic product design approach with higher design flexibility and handles more complexity than the conventional one. Although CAE analysis results may not be perfectly accurate to predict the real application scenarios, but they are good benchmarking tools to ensure the constraint requirements being met for the design. So far, most of the CAD and CAE operations are carried out in an interactive manner.

To perform CAD/CAE operations interactively requires considerable amount of time, efforts and resources for a design unit. Hence there is a need in the product development processes of the industry to work on multiple projects with a generic methodology and a common process model so that changes in the concurrent existing designs can be accommodated and managed. These days, CAD modeling has been established with matured parametric design capability and design intent can be reflected by a set of design patterns defined with user defined features of variable dimensions. Features are defined via parameters of dimensional values and various constraints; and some design parameters are connected to other parameters by engineering concept principles and design constraints [2]. By combining the parameter-based and generative design method [2], also called parametric design, with knowledge embedded computer programming, design processes can be automated by modeling the semantic relations among parameters involved in building various design features with modular CAD procedures. Since a design process usually follows a set of specified steps each time, a computerized reusable design process model can be used to tackle different design applications. To do so, it is essential to create a coded program to automatically generate the required CAD models. Further, in a similar manner, some CAE analysis processes can be programmed as well, such as mesh generation.

Most of these tools the focus is usually on either CAD or CAE application separately and lacks complete potential to handle the other. In most of the time, a design engineer has to work with two or more independent software packages for modelling and analysis and yet has to maintain the information dependencies by checking the constraints applied throughout the engineering design processes. However, due to the tedious dependency relations and the lack of management tools, it is difficult to track and hence lose model details and associated information. Thus it is desirable to integrate CAD and CAE in order to complete every design cycle effectively. In modern CAE software tools, the geometry generated in CAD can be directly taken as input for analysis but the complex geometry has to be modified and simplified in order to get an effective and quick result. Further, the data flow process, i.e. from CAD to CAE, is a one-way translation and the loss of information during translation from CAD to FE model, such as constraints, non-geometrical entities, and geometric intersection details, is very hard to keep track.

Some of the major issues involved in CAD and CAE integration are summarized as follows: (1) information losses; (2) compatibility issues between data structures; (3) breakdown of dependencies; (4) lack of reusability of knowledge; (5) the conflict of complex geometry and its analysis simplification requirement; (6) loss of design and modelling related expert knowledge; (7) difficulties in automation of the design process; (8) unacceptable time associated with the total design cycles; (9) geometry simplification of CAD model and the conversion to FEA model for mesh generation and analysis.

Many efforts have been made to work with standard file formats such as STEP to convey the design intent along with CAD. Some of the issues with this technique are that the generation of STEP files that takes considerable amount of translation and repair effort. On the one hand, some semantically associated data in the CAD model gets lost; on the other hand, if the designer does not want to pass on all the design data along with the model as the intellectual property is concerned, he or she would be difficult to manage with a standard data format. It is advantageous to use parametric modelling for purpose of isolation and control of design data from various computer models.

Therefore, a method of integrating CAD and CAE at parametric level is proposed with the help of a common data structure sharing information with all the involved computer models created by CAD/CAE tools. This data structure is hereafter called the Common Data Model (CDM). Rather than integrating various individual software models, such as CAD and FE mesh models, in a one-to-one "pair-wise" specific interfaces, all the parametric information associated with the design process is integrated in the CDM, a neutral data model managed outside of those specific commercial software environments.

This research involves creating the CDM by embedding engineering concepts, expert knowledge and design standards into a coded computer program and automates most of the product development process with minimum user involvement. By combining and automating parametric CAD and CAE generations, the efficiencies of both processes can be increased and hence considerably reduce the design cycle time. To deal with the geometry complexities and the overall time required for detailed CAE analysis cycles, a dual-loop design process is adopted. In this approach, during initial design stage, the skeletal mid-plane conceptual model for the design object is created, for which, in CAE analysis, the mesh model comprises of only 2D elements thus the calculation takes much less time as compared to a detailed 3D mesh model. The approach of mid-plane model analysis is commonly applied for designing products or parts with shell geometry, i.e. having uniform thickness and material distribution. After the basic structure of the design object is finalized using the mid-plane model, the design then enters the next loop, a detailed design process, in which the design object is modeled as 3D solid with all the design details; hence, the CAE model associated is similarly made of 3D FE elements. Because the major design structure has been validated during the initial mid-plane stage, the detailed model requires less number of iterations to check and can be focused on those sensitive and localized areas. Note that as practiced in the industry in the initial stage, the mid-plane CAE analysis replaces usual detailed analysis with many cycles of iterations, the effectiveness of the overall approach increases and also the total product development time required decreases dramatically.

Proposed research method also uses a design process model that uses CAD/CAE journal files to minimize the overall program development time for generative CAD and CAE processes and to reduce end users' interaction effort as much as possible during each development cycle. Ideally, with the help of journal file recording function of the software tool that records the designer's on-time interactive effort with embedded design intent, coupled with the parametric modeling capability, modular re-executable programs can be derived from such journal files, making sure that the model and analysis development follows the designer's practice and steps.

Thesis first outlines the previous work done in the related research area. Next section provides details of CDM along with CAD/CAE integration process. Further to that, the thesis includes detailed description of the program development for generative CAD and CAE using a journal file. After that the entire process is implemented in the design process of two-phase oil gas separator.

Chapter 2
Literature Review

2.1.Conventional Design Process

Conventional design method for product development, especially for pressure vessels, uses theoretical and analytical engineering calculation-based techniques. The major limitations [1] associated with conventional analytical approach are,

1) Insufficient information on localized stresses and specific stress concentration locations;
2) Over simplification of the actual problems. In order to apply the theoretical engineering calculations to the design problem, the design object is often simplified and several assumptions have to be made to fit it to the endorsed calculation methods that are often regulated via standards and codes of professional institutions and government agencies.
3) Manual calculations. Such traditional approach can involve errors and is considerably time consuming.
4) Lack of optimization for major design governing parameters, such as vessel diameter, following the analysis results. Most of the analysis was done on the macro level, thus it is difficult to use optimization for parameters such as thickness of the vessel and material distribution.

The process of product development possess revolves around engineering design of the required object as design is majorly responsible for various stages of the product life cycle. Product design is a repetitive process which incorporates the specific working and decision-making steps as shown in Figure 2.1.

Initially for the design of any given product requires a fully realized problem definition with clear understanding of all the design related inputs. Also the designer has to be familiar with the appropriate design steps required to tackle the given problem. To finalize the product, decision making steps are required. If the results are not compliant with the requirements, certain steps of the process need to be repeated to get better results. In order to save cost and reduce development time it is required that these design loops should be as small as possible.

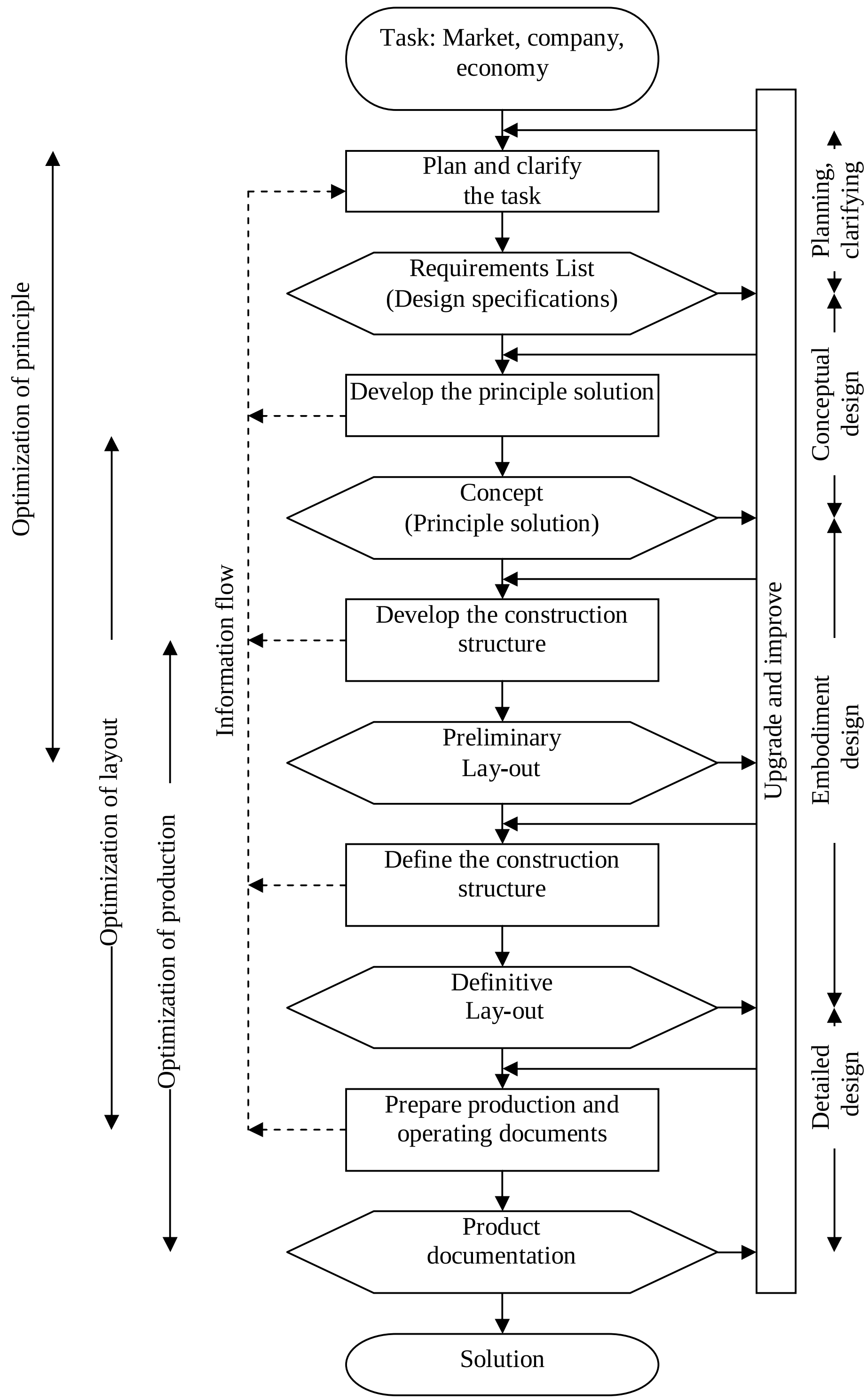


Figure 2.1: Product development process by Pahl and Beitz [3]

Tomiyama et al. [4] made a thorough review on various current design theories and methodologies for product development. The study had also identified the insufficiencies of those methodologies. The drawbacks include lack of considerations for increasingly complex operational and geometric requirements, multidiscipline collaboration, management of complex product development processes, and information integration of various advanced IT technologies for computer-oriented design methods. Globalization trends require advanced virtual engineering and collaboration methods.

CAD technology coupled with CAE offer an effective cyclic product design approach with higher design flexibility and complexity than the traditional one. Although CAE methods may not be perfectly accurate to predict the real application scenarios, but they are good benchmarking tools to ensure the constraint requirements being met for the design.

2.2. Parametric Design and Features

These days, CAD has been established with matured solid modeling technology where parametric design for geometries is supported. In addition, design intent can be reflected by a set of design patterns defined with variable dimensions where patterns are defined via parameters of dimensional values; and some design parameters can be dependent on other parameters using relations [1]. This is the generalized concept for parametric design and modeling. Anderl and Mendgen [5] did some early study which is limited to 2D sketch along with discussion on the advantages and disadvantages of parametric modeling. Susca et al. [6] reported their semi-automatic configuration of 3D racing car design; their method used CAD integrated with a knowledge base for design process with the help of GUI and parametric design. Hoffman and Kim [7] discussed the limitations of parametric modeling, i.e. the issues of under and over constrained models in CAD, and suggested an algorithm to compute the valid ranges of parameters and the number of required constraints.

Feature concept as an abstracted template type of semantic meaning in a design model has been developed using geometric dimensions and other design related data. When one of the feature parameters changes, it also triggers modifications of other parameters and feature elements driven by the associated constraints accordingly. Current feature technology supports a well-defined form of parametric design approach [2] where the physical models of design geometry are coupled with dynamically associated behaviours through the well-known object-oriented software technology and each of the feature type has some generic and modular design semantic concept associated and commonly accepted in the product engineering field. Zhu et al. [8] proposed a feature-based method automatically calculating and validating the parameters involved in the design of a car body. As parametric modeling allows manipulation of model data on a micro parametric level, an automated modeling process using software API can thus be effectively used to propagate changes made in parameters to the specific area of the design object. Major advantage of working with parametric modeling is that most of the commercially available software tools have parametric modeling module coupled with software API. Thus parameter based techniques can be employed to work with number of software tools. Also since parametric modeling can be easily coupled with software API, automatic creation of computer models is comparatively more convenient.

Use of engineering knowledge embedded in the software tool simplifies automatic creation of design parameters and also makes it easier to automate the entire design process. Thus knowledge embedded system along with parametric modeling offers better control over the automatic creation of computer models. Monedero [9] discussed the earlier work done on parametric design and integrating design methods. The research also provides basic definitions associated with parametric design and modeling. Preliminary author discussed problems associated with integrations such as lack of appropriate instruments to modify interactively the model once it has been created and going back and forth between design processes. Myung and Han [10] proposed framework of design expert system composed of a commercial CAD and an expert system based on the design knowledge-base. Work includes development of a method for integration of preliminary configuration design and the detail CAD design. The parametric design expert system is based on design unit concept and the design knowledge base. Author makes use of both API and GUI for the complete process along with parametric modeling and assembly. Zuo and Li [11] applied object oriented terminology applied to CAD. They made use of C++ for programming with AutoCAD for drawing but study did not include 3D CAD or CAE analysis. Author proposed integrating CAD with design knowledge using parametric modeling to improve efficiency and quality. Hauser and Scherer [12] used an approach to intelligent CAD. Major characteristic of this approach is the introduction and the distinct modeling of the strategic, tactical and reactive levels of the design process with a distinction between anticipation of the design approach.

2.3. Advanced CAE Technology

CAE being one of the widely used techniques during design process, various efforts had been made in order to improve the performance of the existing analysis software package. Henrik [13] first explored the limitations for the application of FEM during general practice. He described the problem as "knowledge gap" which is the lack of necessary simplification, strategies and inconsistent and insufficient constraint information and management. Wriggers [14] worked on intelligent support for the FE analysis for automated process of meshing and analysis. Pinford and Chapman [15] proposed "design analysis response tool (DART)" to use knowledge based engineering (KBE) in order to automate the FE model creation process. The major objective behind this was to reduce the time associated with creating and modifying a FE model. DART utilizes engineering knowledge and also has user interface to make changes into the model at any point during the process. In the end it produces a model which is in an acceptable format for an analysis package. Author also focus on simplification of model transformation from CAD to FE and the associativity between the two models. Novak and Dolsak [16] devised a design advising system based on finite element analysis results. Li and Zhang [17] proposed a generic model for intelligent CAE system with the help of model based reasoning.

Major objective was for a CAE system to represent an engineering object with the best conformity to the user's own mental model.

2.4.CAD and CAE integration

Since a CAD system has the various tools to model product geometry while CAE need product geometry as the input for FEA analysis, under the pressure of cycle time for product development, there have been numerous efforts for integration and automation of the computer design and analysis process. Ideally, CAD/CAE integration can be achieved via geometry information sharing and derivation throughout the product evolvement with constant changes. However, historically, the integration of CAD and CAE has been a great challenge in the field of engineering informatics. Some early efforts in model and analysis integration involved automated conversion of CAD to FEM. Yip et al. [18] focused on a *knowledge-intensive CAD (KIC)* which includes integration of design lifecycle and engineering knowledge with CAD, including CAE results; but they did not show how these two aspects interact automatically. Anumba [19] did some of the early work to explore the advantages of integrated CAD systems within a structural engineering context. He discussed and explored basic difficulties associated with integrated CAD along with proposed data structure for bidirectional coordination of graphical and non graphical information. But the scope was limited to CAD only and was one of the preliminary studies. Shephard et al. [20] developed a method to support *Simulation Based Design* via CAD model simplification and data management. It seems the modular design environment works well in a controlled interactive design and analysis setting, but is not clear how the associative design and analysis parameter relations introduced by engineering constraints are maintained consistently. Schreier [21] discuss development of CAD and CAE software tools towards each other and the trends of the software vendors to close the gap between them. He also discusses the benefits of CAD CAE integration along with the benefits of compatibility between various CAD and CAE software tools. The article also describes various new developments in software tools such as "CAD embedded analysis". The major objective of author is to describe the ease of associativity between modern CAE and CAE software tools.

In order to integrate information between CAD and CAE, a middleware development approach is also favoured widely. Propagation of changes is also managed by optimization methods and embedded knowledge. Van der Velden [22] developed a GUI based system called iSIGHT-FD which manages the computer software required to execute simulation design process. It propagates the changes in CAD automatically and changes analysis along with Meshing of entire CAD model without any geometry simplification. Author proposed parametric CAE output using platform to utilize multiple CAD and CAE software tools. Foucault et al. [23] addressed the mesh quality enhancement in conversion of CAD model to finite element model for analysis. Xu and Chen [24] developed a fully automated product design system with CAD- CAE integration and multi-object optimization. Authors employed integration of FEM and iSIGHT optimization for decision making in product development of simple objects. The major disadvantage of method is to develop a complex optimization algorithm along with soft code for FEM and CAD all of which has to be product specific thus the study to the study is good for preliminary simple design but the system is very hard to be modified for more detailed and complex engineering problems.

2.5.The Current State of Art for CAD and CAE Integration

Some of the earlier efforts manage considerable automation advance as well as well association of model information. Wei et al. [25] proposed automatic generation of finite element analysis using the ontology based approach by defining the fundamental analysis modeling knowledge into a set of formal ontology. Aziz and Chassapis [26] developed a knowledge based system for integrated engineering design process from the initial concept to production using the feature based modeling and design for CAD and FE analysis. The system utilises manufacturing and design knowledge bases. Chapman and Pinfold [27] discussed limitations of traditional CAD and advantages of using KBE along with FE analysis with the help of a “concept development tool” for efficient organization information flow and as architecture for the effective implementation of rapid and iterative design solutions. It was one of the initial studies to enhance the capabilities of existing CAD and for knowledge utilization and sharing. Colombo et al. [28] addressed the need of software to support engineers in complex design and proposed KBS tools. Author also suggests a conceptual framework and philosophical approach for classification of knowledge types along with relationships between various functions using mathematical model to define ontology. Xu and Wang [29] proposed using of *Multi Model Technology (MMT)* for integration of CAD/CAE/CAM. MMT uses object-oriented technology (OT) into the product modeling process together with feature-based modeling technology. It includes use of a single basic solid CAD model to generate all the other required models in the subsequent layers of product development process. Features are defined to maintain the associativity between them and feature manipulation is used to maintain integration between CAD and FEA models. In application studies, Yan and Jiang [30] proposed an integrated method of CAD/ CAE /CAM for the development of duel mass flywheel. The interference between various software tools is obtained by using a uniform product data model. Method used specialized software tools to evaluate analysis such as welding strength. One of the major issues involved is that the system uses a large number of software tools and thus making interoperability and exchange between models complicated.

Most of the work done before was to develop initial product model and lacked recursive nature of an actual design process. Albers et al. [31] proposed a strategy for the development of engine crankshaft with the integration for CAD, CAE and genetic algorithm. A Java based interface is used to integrate CAD and CAE. Genetic algorithms are used as optimization tool along with graph analysis. But authors do not propose any means to complete the design loop. Cao et al. [32] developed a middleware to transform CAD models into acceptable CAE mesh model, i.e. HEDP (High End Digital Prototyping). It can manage model simplification and defeaturing of CAD models to make it acceptable to FEA meshing and also get quick results; but the integration is one-way traffic and lacks the recursive loop support. As during a design process, the object to be designed goes through multiple design loops before being finalized, thus the designer needs to recalculate the parameters involved several times. For effective integration of CAD and CAE, along with parametric modeling, engineering knowledge embedment into the process can reduce the overall time for the design and can incorporate design standards and codes into computer program which can be reusable. Penoyer et al. [33] used KBE along with CAD, CAE and CAM for complete product development. The approach was GUI based with KBE to manage majority of the product lifecycle process. But the authors do make use of embedded knowledge rather it is suggested to use direct user interface thus giving lower automation in the process.

2.6. CAD and CAE Integration Problems

From above efforts it is observed that the major hurdles in the attempt of CAD and CAE integration are information losses, compatibility issues between CAD and CAE software, breakdown of associations, reusability of knowledge, conflict of modeling complex geometry and its analysis simplification requirement, loss of design expertise, difficulties in automation of the design process, unacceptable time associated with the total design process, geometry simplification of CAD and its conversion to FEA model for mesh generation and analysis. Most of the efforts have been made to work with standard file formats such as STEP to convey the design intent along with CAD. Some of the issues with this technique are that the generation of STEP takes considerable amount of work. Also as CAD and the data are inter-linked there is a chance of losing some data. As far as the intellectual property is concerned, the designer may not want to pass on all the design data along with the model which is difficult to manage with standard data formats. It is advantageous to use parametric modeling in case of isolation of design data from various computer models.

2.6.1. Data Interoperability

One of the major difficulties in integration of CAD and CAE is association of design data between them. Data associated with CAD is usually geometric however FE model requires mesh and material related data associated with the geometric model imported from CAD. Hamri and Lèon [34] suggested using *polyhedral* model as an intermediate model between CAD and FE model for interoperability. They recognized the need of re-analyzing the same CAD model multiple times with modifications in the evolution of product design phases. Arabshahi et al. [35] identified the potential and did some of the earliest work on CAD FEA integration. Objective behind this study was easier, more robust and faster transformation from CAD to FEA. Author describes the requirements for an automated system for CAD to FEM transformation with major focus on attribute editing and two way link between FEA attributes and geometry along with feature recognition. Johansson [36] proposed an integrated KBE, CAD and FEM for automated system for preliminary production preparations and for complete automation of the process. Some of the issues that the system encounters are compatibility with available commercial CAD and CAE software tools and difficulties in order to develop complicated models. Su and Wakelam [37] worked on creating an intelligent hybrid system to integrate various CAD, CAE and CAM tools in design process using a blend of rule-based system, artificial neural networks (ANNs), genetic algorithm (GA) into a single environment using parametric approach for model generation and rule based approach to control the design environment. CAD & CAE data model are different, therefore the geometry has to be further processed, e.g. converting to mid-plane model, or simplifying the model, etc. It is convenient for integration if there is a common platform between CAD and CAE for information of both the models to co-exist.

2.6.2. Long Design Cycle Time

One of the major issues in improvement of current design process is to reduce the overall time associated with various design stages. It is desirable to reduce time associated with the process since it reduces the required resources as well as development costs. Resh [38] proposed use of CAE to shorten the development cycle time well as suggested that the process needs to be reformed in order to reduce duplication and human errors. Kagan et al. [39] managed product development using an integrated CAD and CAE software which uses B-spline model in order to reduce the development time, cost associated and simpler fine tuning process for the product. They developed a modeling method in which same B-spline model was used for both CAD and CAE thus eliminating the need for conversion, but the study does not utilise any automation or use of engineering knowledge and standards in process

2.6.3. Design Process With Integrated Engineering Systems

As mentioned earlier one of the difficult tasks in CAD and CAE integration is association of data and managing the models to avoid any loss of information. Numerous attempts to solve this problem used knowledge embedded or knowledge based systems to manage the semantic relationships. Computer modeling coupled with engineering knowledge can manage the entire design process effectively. Zeng et al. [40] proposed a multi-representational architecture (MRA) to facilitate the transformation of information from design models to various support analysis models. The major focus was on ABBs (Analysis Building Blocks) for solid mechanics and thermal systems that generate FEA in order to bridge the gap between design and analysis model. Xu et al. [41] proposed to integrate CAD/CAM/CAE based on CATIA for the end-to-end process in cylinder head development using Multi Model Technology (MMT) to create consistent and associated CAD models. Chen et al. [42] suggested use of Unified feature modeling for integration of CAD and CAx for the process of product development process. The feature was defined as a “relationship object associating geometric entities”. The author utilized knowledge base and unified feature information database for information sharing, consistency and control among different models. The work is primarily based on feature association and unification concepts which include three-level geometric and non-geometric relations along with some of the early work on computer modeling and analysis for complete product development with integration of various computer based technologies and tools along with product data management (PDM) technology. Smith and Bronsvoort [43] discuss various previous approaches for design and analysis model integration and also suggest a new method by integrating an analysis view into multiple view feature modeling using automation and analysis knowledge. Author provides clear discretion of design model and analysis model and their differences. They also discuss the idea of maintaining multiple views of the model a time and modification of views and design by feature conversion. As per their definition, *"An analysis view in the multiple-view feature modeling approach should be a view of the product that is suitable for an engineer to perform analysis with."*

2.7. Feature-Based Approach for CAD and CAE Integration

Feature is essentially defines the basic structures that made up a particular model, thus a model is built up with one or more features as building blocks. In conventional modeling process, initially a base feature is created and the is further enhanced by adding other features or simply adding more details to it until required model is obtained. Development of features various follow the designer's intent and thus are subjected to changes as the design progress. Thus a model based on features can be changed by manipulating the features which in turns can be used to reduce the development time. If features are integrated with parameters and other features, changes made in one feature can be successfully propagated [2] through the entire design. Monedero [44] did some of the early work with parametric design and integrating design methods and provides basic definitions associated with parametric design and modeling. Author also lists problems associated with integrations such as lack of appropriate instruments to modify interactively the model once it has been created and going back and forth between design processes. Deng et al. [45] incorporated the use of feature based modeling and analysis for CAD and CAE integration where various features associated with both CAD and CAE including all geometric and non-geometric ones. The prototype software for injection molded product design tried a feature mapping method for CAE feature simplification such as ribs. Kao et al. [46] discussed the parametric and feature based automatic generation of CAD for thread rolling die-plate geometry to regenerate the model with varying set of parameters and features with the use of an external spreadsheet file to be used as a source of parameters for CAD/CAM/CAE. Though the changes were automatic, interface was GUI using a predefined template. All the related parameters had to be calculated beforehand and needs to be written into spreadsheet and corresponding changes are also required to be made interactively. Chen et al. [47] discussed semantics of design and machining feature and identifying information entities, relations, constraints in each view and further generalizing common entities in order to develop a consistent product information model. Use of features is proposed as information medium in order to integrate conceptual design, detailed design and process planning are discussed along with feature association and unification are described in with relation to unified feature modeling scheme for information sharing and consistency control. Chen et al. [48] made use of unified feature for integration of CAD and CAx models for concurrent engineering for information sharing and consistency control between various application feature models and identifying feature constituent levels for controlling the consistency among them. A unified feature consists of common attributes and methods for all the supported application features.

Individual work on CAD and FEM models has been done mostly with the help of Knowledge Based Engineering. Some of the techniques employed for these individual systems can be brought together along with parametric modeling to improve the effectiveness of CAD CAE integration. Peak [49] described problems associated with CAD and CAE interoperability, fine grain associativity gaps and software tools' limitations such as knowledge modularity, reusability, and accessibility, directionality, fidelity, control, and multi-disciplinary associativity. Work also focused on mapping various attributes between CAD and CAE in order to reduce the overall time and cost associated with design process. Zeng et al. [50] suggested the use of *ZAP*, a knowledge based FE modeling method, to reduce design time and suggested CAD-FEA integration at knowledge level and stressed the importance of automation in idealization of CAD and mesh generation. Lee [51] focused on creating a single model containing both CAD and CAE features and explored the advantages of a *common modeling environment* and *bidirectional* CAD and CAE integration with multiple feature representations and limited automation.

In current research approach, integration of CAD and CAE at parametric level is proposed with the help of a common data file sharing information with all the design models thus called a CDM. Rather than integrating various models themselves in a graphical software tool, all the parametric information associated with them is integrated in a data model outside the graphical environment. Also to get around the tedious job of programming for generative CAD and CAE a smart method of program development is applied with the help of pre-recorded journal files. To deal with geometry complexities and overall time required for CAE analysis, a design process involving duel design loops is adopted.

Chapter 3

Proposed Design Procedure

By definition *"Design is the act of formalizing an idea or concept into tangible information which is distinct from making or building"* [52]. The design process is a repetitive process which incorporates the specific working and decision-making steps. The processes are interlocked due to the evolvement of the state of development and engineering details. Hence, design cycles are inevitable.

Product development requires initial preparations such as collecting information, collecting various solutions, calculation, CAD modeling, drawing generation and evaluation. To finalize the product, evaluation steps are required. If the results are not compliant with the requirements, certain steps of the process need to be repeated to get better results. In order to save cost and reduce development time it is required that these design loops should be as efficient as possible. Also it is desirable to make the iterations effective and timely.

3.1. Parametric Design Process

Parametric modeling [2] allows manipulation of model data and features at parametric level, thus an automated modeling process using application programming interfaces (APIs) can be effectively used to propagate changes made in parameters to the design objects. The major advantages of working with parametric modeling are the systematic control to the engineering design intent and the quick propagation of changes according to new input conditions, i.e. design changes. Parameter based techniques can be employed to work with a number of software tools and easily coupled with software APIs; then automatic creation of computer models becomes comparatively convenient.

The use of engineering knowledge embedded in a reusable software program simplifies automatic creation of design parameters to reflect the specific design intent. In fact, it has been a commonly adopted approach to automate design processes in aircraft and automobile industries. Their experience proves that a knowledge embedded system along with parametric modeling offers better control over the product development processes.

Most of the computer modeling and analysis software APIs are made available for common programming languages. Their entities and related operations can be created and executed via programmed functions and controlled parameters automatically and hence, the software models are “generated” instead of interactively developed by the user via GUIs. This approach is largely known as generative approach.

3.2. Design Information Flow and Sharing

Figure 3.1 show all the stages involved in a product design process. Development of each stage depends upon the relative progress of succeeding and proceeding module. Figure 3.1 also shows the information flow between various processes. The proposed design process is a semi-automated one with the help of generative CAD and CAE techniques using a centralized data repository, i.e. CDM, as briefly introduced in Chapter 1 and in the published article [53], from which all the required parameters are imported. The structure of CDM is to be further introduced in Section 3.3 of this Chapter. In order to make the entire process as flexible as possible, it is essential to keep the information associated with every module in a neutral format for data sustainability.

Modern CAD and CAE software tools are equipped with parametric modeling tools to associate variable parameters with specific design features. For example, a cylinder feature can be created with the parameters of diameter, length, location, orientation; and they can be updated easily as well as the CAD model. Partial automation of the computer modeling and analysis process by using predefined templates can be achieved coupled with parametric modeling for changing design conditions.

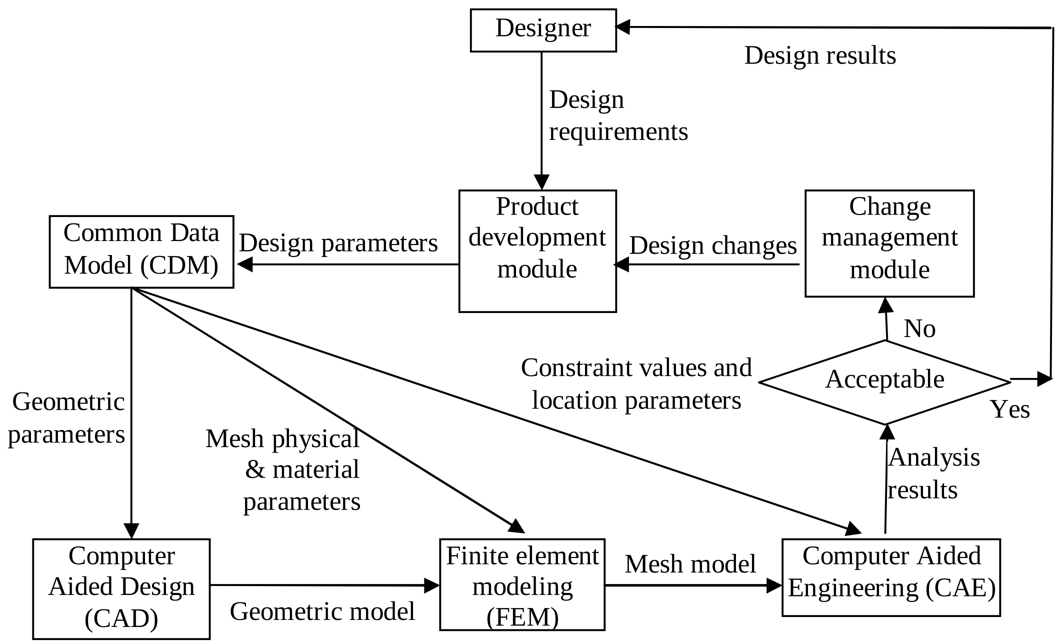


Figure 3.1: Information flow between various stages in design process

Ideally to maintain the associativity between modeling and analysis, it is preferable to use a single software tool for both. Unfortunately, most of the commercially available software tools cannot uniformly support both areas and thus the designer has to work with two or more software models associated with the corresponding tools. Again, ideally, user defined information and knowledge should be represented in a neutral, reusable, and scalable data structure form. In this study the CDM is a text file with design related parametric data arranged in an orderly manner to use it as expression file in Siemens NX. Then for a system with multiple software modules, the user defined data structure can be interfaced irrespective of the APIs of the modeling and analysis software tools. Since the structure does not depend on the type of programming tools used, the structure can incorporate any CAD or CAE software API functions which provide flexibility for selection of appropriate tools.

In industrial practice, typically in product development, many different software tools are used and their native formats are proprietary. Some standard computer models, such as STEP files, are used to translate data among computer tools. However, the issue of information loss in the form of geometric details or modeling expertise etc. is well known in the industry. Usually, the information associated with the design model of each software tool is not only geometric but contains also non-geometric details associated with every aspect of the design which makes a complete product information model complicated to be handles in a coherent way conveniently. This is because of the differences of model definitions associated with different software tools; there are always anomalies to map the transformation of models from one form to another. Thus consolidating the design data from different computer models makes it easier to control the information. That is why a CDM is suggested as shown in Figure 3.1.

The details of CDM structure and implementation details have been introduced in next section. For the ease of comprehension, the CDM can be understood as a central data structure that stores all the user defined parameters related to all stages of a product development as well as the explicit references and constraints among them. CDM allows the outside integration of data that can be easily interfaced among functional software tools, such as CAD and CAE. Due to the uniform definition of the CDM, then the product modeling in each of the software tool deals with only a specific view of the total information, and the user programming for the specific product model becomes much more manageable.

During the design process, CAD serves the purpose of providing the product initial geometry, the visualization model of the product, and geometric inputs for the finite element (FE) meshes. The additional information associated with the mesh elements such as physical material properties of the design object are taken as parametric input from the CDM. The finite element mesh model is then used in the simulation environment. The model is then applied with the required loads and constraints in order to carry out the numerical analysis calculations. Similarly, such additional information associated can also be extracted from the CDM. Thus by using parametric input from a neutral CDM, consolidation of user defined product model with specific information for different software tools can be obtained. In a computer program, association of specific data from the CDM to a modeling or analysis tool can be conveniently achieved by data and file handling functions.

3.3. What is CDM?

Following the conventional design process and standards, after starting the design project, the user is required to input the "*design requirements and specifications*". Then based on the product development knowledge, all the design parameters at the engineering conceptual design level are determined and embedded into the conceptual design models with CAD tools. Engineering analysis with CAE tools is conducted to verify the design concepts at different abstract levels and from different aspects.

It is the proposed concept that a centralized parameter repository that contains those driving design and analysis parameters as well as their explicit constraints is developed as a CDM such that the characteristics of conceptual design and the CAE analysis settings are kept in a systematic form and can be managed for their consistency. In other words, the CDM is made of all the design semantic parameters required to build CAD model, FE mesh model and to conduct engineering analysis with the assistance of knowledge-based tools and software APIs. Figure 3.2 shows the basic concept of data management system with CDM.

The proposed design procedure using CDM to integrate CAD and CAE is based on two corner stones, i.e. parametric design and analysis in CAD and CAE environments respectively, and the change management strategy that support a progressive and cyclic approach for the iterations of product conceptual and detailed design evolution. CDM can be captured in a data structure and stored in a neutral data file at the same time. The design data model generated this way can be recorded, documented and rationalized along with the different phases of a product lifecycle.

Hence, this CDM is a dynamically managed data file and its contents are incrementally created and updated. Hence, the engineering intent embedded is detailed gradually in stages over the cycles of design consolidation. The details of CDM structure is explained further in the following portion of this chapter. Based on the CDM, a design program further calculates and/or selects all the required geometric and analysis parameters according to the standard industrial design procedures and the required regulatory codes.

The content in CDM is of three general categories (1) geometric parameters; (2) Non geometric functional parameters; (3) intermediate design related parameters. For the purpose of examples, such parameters for the design of two-phase vertical separator design are shown in Figure 4.1.

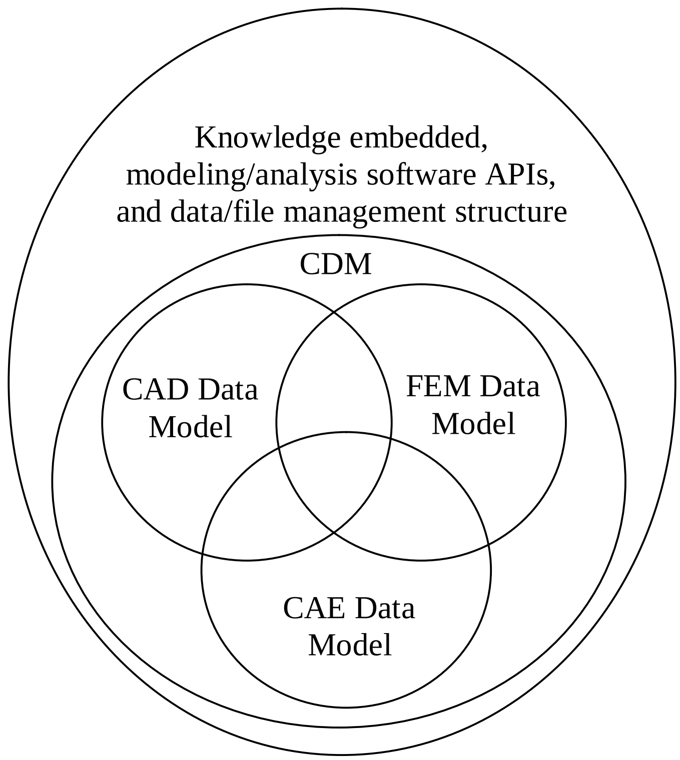


Figure 3.2: General working aspects of CDM

CDM assists in retaining all the information at a centrally organized data structure and acts as a system “switch board” for all the parametric input to CAD, FE and CAE user-defined interfaces. Information in CDM about parameters not only contains the numeric values but also the specific units associated. All the information is carefully represented in a specific data structures that can be directly transmitted to and from the used program API functions and arranged into the readable with concise comments or instructions. Note that the arrangement of information in the CDM and its data file can be customized depending upon the software tool requirement. For example, the Siemens NX 6 parametric expression input file requires information in the form of *"[UNIT] NAME = VALUE"*. For other data exporting and importing purpose, depending upon the software input requirement, the intermediate file format of the CDM can be manipulated with data and file handling algorithms, such as using ".XML" format. In current study, the data file is first generated as a text file (.txt) and then converted into an "expression" (.exp) file which is used as parametric input for the NX software tool.

To enable the mechanism of managing design changes using CDM in a parametric approach, functional programs are created so that CDM parameters are associated and supported with the either automatic routines or interactive user input. The program functions also keep the CDM structure and contents consistent and updated. Thus the CDM management system is always associated with the CDM data file which serves as a permanent repository; and a set of fully functioning, parametric and knowledge-driven design and analysis functions are developed and invoked in the background from time to time.

The comprehensive CDM supporting all the models involved in a design needs a large number of parameters used throughout the development process and they have to be supported with tedious and repetitive input operations. More development has to be done if the proposed solution is to be used for any practical engineering project. In order to represent entire design data structure in terms of parameters, the CDM has to represent:

1) Meaningful design records by versions of CDM files each of which includes embedded design intentions collectively as a "state" of the design evolution;
2) Those referenced common parameters that constituent the required constraint and interference relations among different operations such as CAD model and FE mesh model development;
3) The complete data set to ensure the robustness of reconstruction;

4) Independent and neutral manipulation capability interfaced with CAD and FE software system;
5) Data system which can be easily understood by engineers and associated with respective aspects and features in the model.

Structure of the CDM shown in Figure 3.3 follows the proposed design procedure. Initial entry to CDM contains design and operational requirements provided by the designer. These inputs are then used to first calculate non-geometric intermediate design information like additional operating parameters as introduced in the section of "Design calculations" previously. The first set of complete parameters to build CAD model and CAE analysis are generated for the conceptual design.

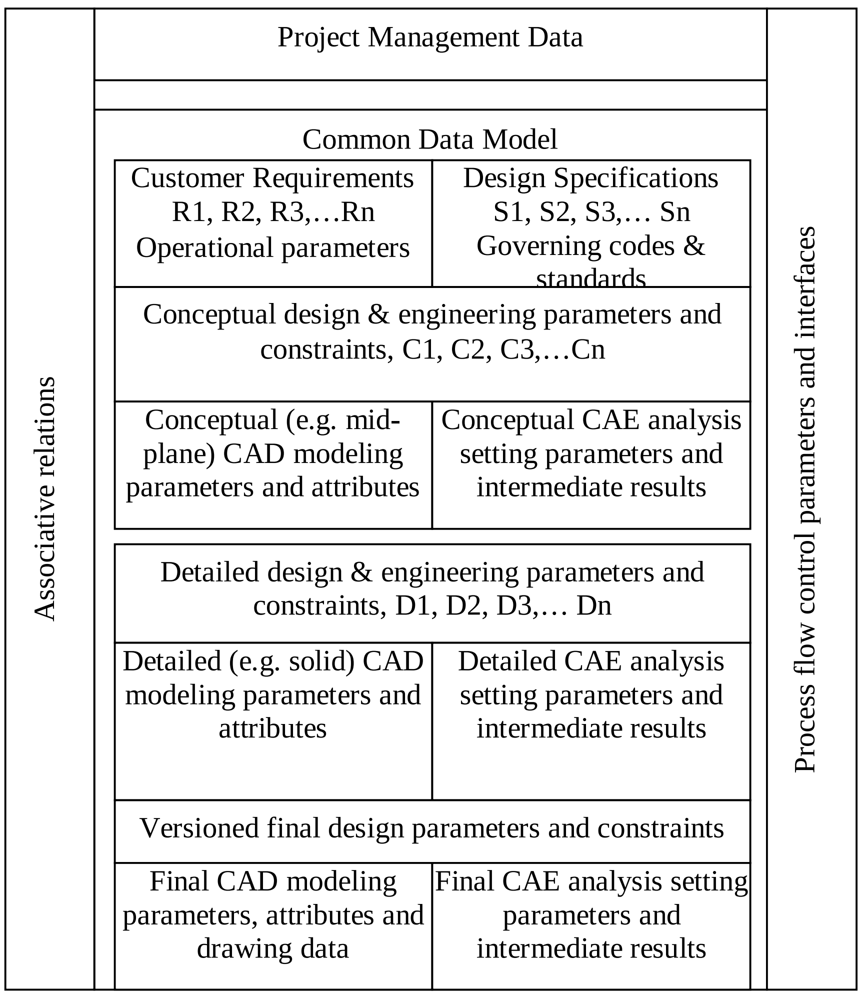


Figure 3.3: Structure of CDM with the progress of design process [53]

These parameters include geometric dimensions, design parameters and limits and constraints required to generate mid-plane CAD model, and FE mesh, and to apply CAE analysis. This conceptual design cycle is repeated until the desired result is obtained. In the next phase, all the parameters required for the detailed design such as the dimensions for the internal components and constraints are generated automatically using the program code. These parameters along with previously refined parameters from conceptual model provide required parameters for the final solid model.

This final list of parameters is then used to generate 3D solid CAD and FE model and the final CAE analysis. These final parameters are then refined with each design iteration till designer’s intents are met. The final parameters are the versioned refined parameters which can be used to create a final model further used for manufacturing.

With a similar software setup at each workstation, a commonly placed data file can be used in collaborative environment by a development group. Coupled with optimization algorithms and detailed manufacturing embedded knowledge, CDM can be further refined to include manufacturing details such as tolerances.

3.4. Proposed Design Process With CDM

CDM acts as a kernel data structure connecting all application programs, supported with file and data management software modules that control the user interface as well as the information security and integrity. Ideally, there should be a system for all applications to be combined in a single interface. Also the system should be open, or to be able to enhance by adding new application or taking advantage of improvements in technology without invalidating previously obtained data.

The basic purpose of this study was to develop an efficient and time saving design development procedure. Ideally, a fully integrated design method can be developed to automatically generate all the engineering information, computer models and analysis results. To be realistic, such a method could exist for those products need the reuse of a well-established and generic design procedure with a constant designing methodology like pressure vessels. A knowledge driven and parametrically generative approach is taken for both CAD modeling and CAE analysis interfaces.

Just like normal design practice the process, as shown in Figure 3.4, is divided into a number of stages which follow similar logic in the program code. The design processes are arranged in such a manner that they flexibly follow the conventional as well as advanced, integrated, and parametric product design methodology. In the proposed method, CAD and CAE integration happens at parameter levels of CDM.

During the initial stage, the customer's requirements and technical requirements are accepted by the designer become the initial input data of CDM. These inputs are evaluated and used in the engineering knowledge-embedded calculations of a product model generation program. The equations specifically used for a two-phase oil and gas separator design, are to be introduced in Section 5. In this stage, the CDM serves to record the driving design parameters and key constraints. Note the product model should be generated with the help of conceptual engineering knowledge, industrial standards and codes through object–oriented programming coherently.

To explain the product development process with more insight, In Chapter 6, an example of pressure vessel, i.e. a two-phase oil-gas vertical separator is used throughout the thesis as the case studied.

To make a preliminary quick evaluation of the developed design concept, an abstract CAD model is created automatically by using the necessary APIs and parameters in CDM. Next, the CAD model generated is used as the geometric input to create a finite element mesh model. FE model utilizes meshing and material parameters from CDM automatically to assign mesh physical and material properties to the model. Since the model has a planar CAD model as base, the elements generated are also planar (2D). In case of design of a thin vessel like separator, the mid-plane for the design is the shell layer geometry at the middle of the thickness of the designed vessel for the initial calculations.

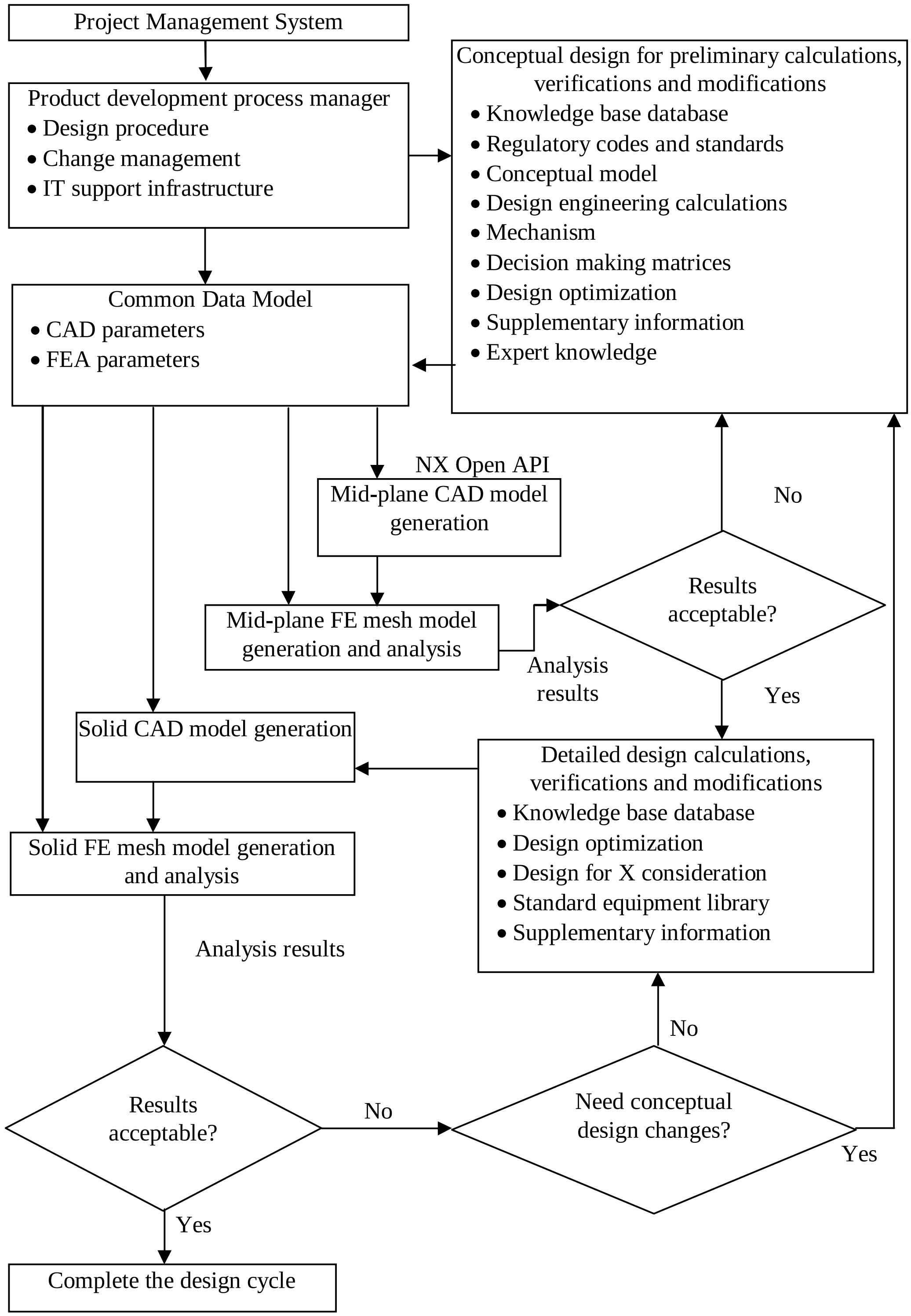


Figure 3.4: Proposed design process with CAD/CAE integration [53]

The mesh model thus created is then used for computational analysis of the model in a CAE environment. During analysis condition specifications, such as load values, can be directly inserted from CDM via programmed functions. Also all the constraints associated with the analysis can also be stored in the form of embedded knowledge specific to a particular design object and applied to the analysis model. Results obtained from the analysis are then provided to the user to make a decision for the validation of the design and to make any further change if required.

If the design needs further changes to fit the requirements, the designer is given multiple options based on general change scenarios and then new changes are incorporated into the CDM. Then changes are implemented by re-running the model generation program to update all the parameters associated with the design. This new set of parameters is then used again automatically to create the model for analysis through the mesh generation and CAE setting programs. After that, a new set of results are provided to the designer for verification. This conceptual mid-plane design cycle continues till desired results are obtained.

Once the conceptual design stage is completed, detailed design phase kicks in. Usually, to enable more detailed analysis and finalize the design model, a much more detailed solid (3D) CAD model is generated using the detail design parameters predefined in the CDM. Similarly, CAE model has to be detailed to fully reflect the features of the new geometry defined. So, a solid (3D) FE model (solid mesh model) is created based on the solid CAD model. This FE model is then used for final numerical analysis in CAE software. Like the mid-plane phase before, the constraints and their locations are built into the user-customized CAE analysis program in the form of embedded knowledge. The change management is similar to the conceptual design stage.

Detail design iterations occur through the designer's evaluation and automatic propagation of changes to all the parameters in CDM and subsequently to all the computer models. Updated changes can be automatically reflected in both CAD and CAE models by executing the associated generation programs again iteratively. This iteration continues until satisfactory results are obtained. Depending on the intention of design changes, the process can be rolled back to either conceptual design modelling or detailed design modelling stage.

The final detailed design parameters are extracted from the CAD model; and they are recorded in the CDM and provided to the designers as well as other users via different required output formats. During each stage, the CDM information associated with every cycle is stored separately to maintain a history of design revision cycles. Since computer models can be automatically generated for any stage cycle with the help of CDM, models of a particular stage can be reproduced automatically through API programming. Hence, potentially, many configurations of a product family can be derived for comparison or mass customization purposes. Further, such a system enables user to input desired standard sizes for various components which then the program code uses during the automated design process.

In a design process, the change in scope calls for a considerable increase in the engineering efforts due to the dependencies of associated features, while the changes in the sequence of feature creation impose even more complicated "patching" work. Most of the efforts associated with design changes are towards re-engineering and remodelling in order to adapt with user requirements. Each change made in the geometry of the design equally has to be propagated to CDM and all the other aspect of the design. In order to successfully propagate the changes throughout the process, the modeling and analysis software, i.e. a knowledge-based generation module in the prototype package, should be developed with logics and functions capable of supporting the changes. More research work is to be expected to work out a generic solution.

Depending upon the availability, proposed method can accommodate any programming software tool and CAD and CAE software tools with API capabilities. In Appendix C, Figure C.1 gives the pseudo-code structure of the overall proposed design process. Figure C.2 shows the CAD generation process while Figures C.3 and Figure C.4 show the pseudo-code structures of finite element meshing and CAE analysis processes respectively.

Chapter 4
Common Data Model

As described earlier in the product development process, CDM comprises of all the design related parameters. CDM is essentially a data file which contains design information in the form of design parameters. CDM also assists in retaining all the information at a single place and acts as input to computer software API to automatically create all the required computer drawings. Information in CDM about parameters not only contains the numeric value but also the specific unit required for API input. All the information is carefully exported and arranged into the data file using a computer program to make it suitable for API input. The arrangement of information in the data file may change depending upon the software tool requirement.

4.1. Construction of The CDM

For current study a case for the design of two phase oil-gas separator is used and from here onwards this case is used for explanation and elaboration purpose. The example separator is designed with the reference to industrial practice [54] and according to American Society of Mechanical Engineers' Boiler and Pressure Vessel Code (ASME Code) section VIII. Pressure vessels are designed to withstand the loadings exerted by internal and external pressure, weight of the vessel, reaction of support, and impact. Temperature, pressure, feed composition and its mass flow rate is considered to select type and design of vessel and to come up with the dimensions of vessel.

4.1.1. Engineering Calculations

As usual, engineering calculations have to be dealt with systematically. In this work, they are embedded by programming them into a procedure when initiating the CDM based on the customer's requirement and the designer's specifications.

They are taking as constraints. Although there is certainly a room to make them in a more organized and modular system structure, due to the limitation of time available for this particular work, they were simply implemented in a C++ code program.

For this particular study the only load considered are internal pressure and temperature. Vessel size is decided depending upon the flow rate requirement. Table 4.1 gives the data structure suggested for the CDM for the design of a two phase oil-gas separator.

Table 4.1: Typical parameter data structure of CDM

CDM parameter name	Engineering parameter symbol	Definition	Unit
Customer Requirements			
R1	*dm*	Droplet size in fluid to be separated	μm
R2	*Ql*	Liquid flow capacity	BPD
R3	*Qg*	Gas flow capacity	MMscfd
R4	*tr*	Retention time	min
R5	*μ*	Viscosity of the fluid	cp
Design specifications (Including input constraints for finite element analysis)			
S1	*P*	Operating pressure	psi
S2	*T*	Operating temperature	°F
S3	*API°*	Specific gravity of oil	-
S4	*SGg*	Specific gravity of gas	-
S5	*Ss*	Material Strength	psi
S6	*E*	Joint efficiency	-
S7	*W*	Vessel weight	lb
Parameters needed to build conceptual design model			
C1	*D*	Shell diameter	in
C2	*Lss*	Shell length	in
C3	*Hh*	Head height	in
C4	*dnx*	Nozzle diameters	in
C5	*dnlx*	Nozzle location dimensions	in
C6	*sx*	Support dimensions	in
Additional parameters associated with construction of detailed design model			
D1	*tx*	Component thickness	in
D2	*idx*	Inlet diverter parameters	in
D3	*vbx*	Vortex breaker parameters	in
D4	*MEx*	Mist extractor parameters	in

4.1.2. Generation Of CDM Using Analytical Design Process

The three basic information sets used in the program to generate the data model are geometric information, engineering rules, regulatory standards.

CDM has to be made in such a way that it should be simple, without any unnecessary details but it should still posses all the required information organized by sub-domains of the design scope and stages. Since the entire design process reliability depends upon CDM, information in CDM should be accurate and every parameter must be exhaustively mapped with those corresponding related feature. As explained in the design process, CDM information is used to construct 3D solid as well as mid-plane representations of same design object. Since CDM aides in automatic CAD and CAE integration at parametric level, the content of the CDM and the data file itself is generated and managed automatically via a customized and integrated design and analysis system.

Figure 4.1 shows the partial data structure of CDM for the design of vertical separator design. There are various parameters associated with design with some parameters getting used in multiple applications as highlighted. Most of the parameters are refined during conceptual phase and are used in detailed design phase along with additional details for full representation.

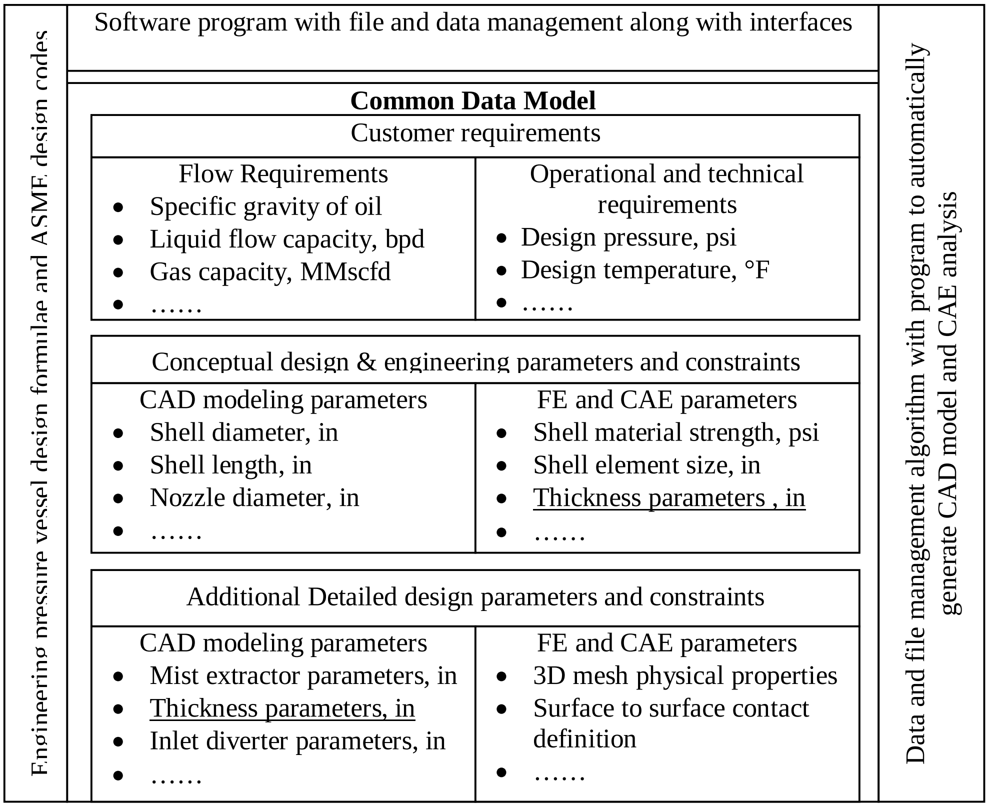
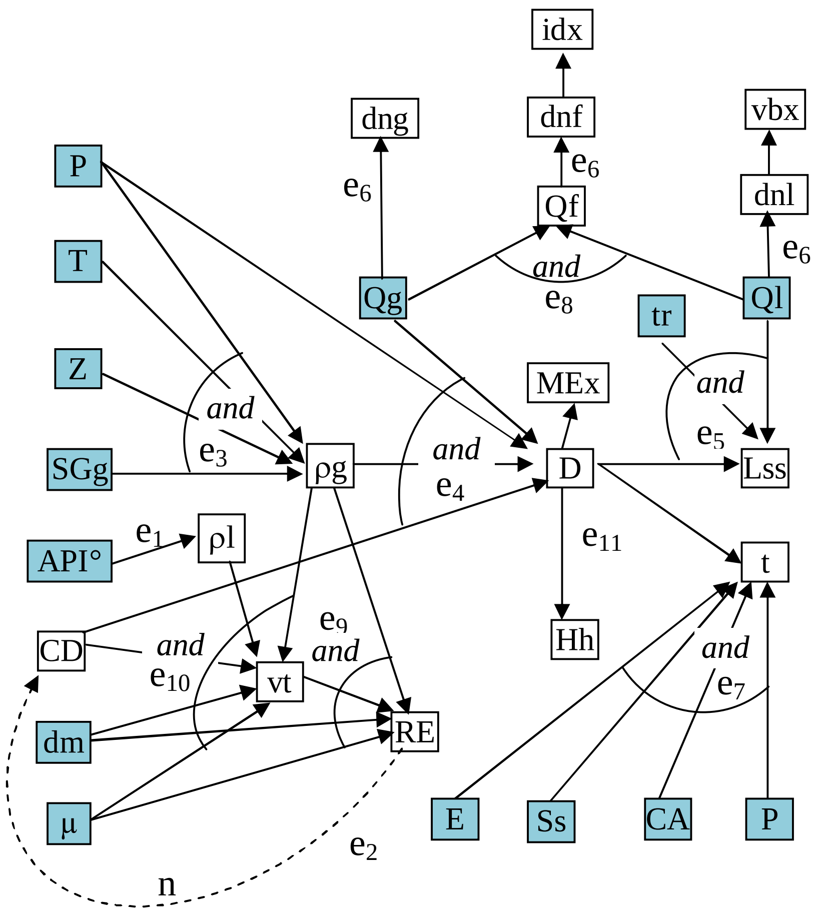


Figure 4.1: Partial CDM data for vertical separator design

4.2. Parameter Semantic Map of CDM

As explained earlier, CDM is a collection of parameters which, in turn, forms all the features in the design. One of the key properties of a design is that all the parameters are more or less interdependent and are semantically connected to each other in the engineering calculation program. To maintain the associativity between all the parameters involved in the design throughout the design process, the relationship between them needs to be identified. As shown in Figure 4.2, dependencies between various parameters affect the development of the design as well as propagation of the change. Change in any one of the design parameters has an impact on rest of the design. Moreover, the relationship between various parameters is maintained through embedded knowledge in the computer program.



List of constraining equations:

e_1 : Eq. (A.1)
e_2 : Eq. (A.5)
e_3 : Eq. (A.4)
e_4 : Eq. (A.8)
e_5 : Eq. (A.9) and (A.10)
e_6 : Eq. (A.18)
e_7 : Eq. (A.16) and (A.17)
e_8 : Eq. (A.19)
e_9 : Eq. (A.6)
e_{10} : Eq. (A.7)
e_{11} : Eq. (A.11)

Notes:

- Symbols in shaded boxes are known parameters
- Refer to the nomenclature for the definitions of the parameters.

Figure 4.2: Parameter semantic map for vertical separator vessel

The semantic links of the parameters in CDM follows certain patterns in accordance to the established design procedures and codes such as those laid down for pressure vessel design under ASME codes. The first set of parameters which are the user input and the known values, govern the calculations for the parameters associated with conceptual mid-plane design. Parameters associated with detailed design depend on all the parameters in the conceptual design as well as input parameters. Various parameters are entered into the CDM in the same order as per they are created automatically along the process.

However, CDM alone is simply a data file of some sort with all the data thus in order to associate information in it and to establish an automatic semantic relationship, a set of program interfaces are essential to mange those constraints or dependencies. The nature of semantic relations depends upon the nature of parameters and their application in forming various design features. Progressively, it can be appreciated that the initial sets of parameters in the form of user input are of functional and operational type, for example they also include parameters associated with material properties. With the help of design intelligence such as those related to flow and operational relations in pressure vessels, the initial set of parameters then produces the intermediate set of mixed functional and geometric parameters. Finally through engineering calculations and selection from standard sizes, a final set of geometric and functional parameters is generated which are used for model generation and analysis.

4.3. Dynamic Data Flow and Change Management

In this separator case, after every modeling cycle, the user is given the simulation analysis results. The static simulation results usually produce maximum stress and maximum deformation values with locations. Depending upon the standard used, quality check requirements and design practice, by checking the simulation results, the user can choose to change the design. Program code used to build CDM and design models further gives the user several options to make changes to design. Any change to design has to first take effect at CDM since it governs all the further process. Options for changes include operating conditions and material. Input from user is taken as a trigger to re-run the knowledge embedded program code to recalculate all the values again and are stored in CDM again. Since CDM is a data file in a generally accessible format, the user may choose to make changes directly with the file. As explained earlier, all the parameters in CDM are semantically associated with each other therefore changes made through the program code will always end up modifying all the related parameters.

In order to change a specific parameter like feature thickness without changing any other parameter, it is convenient to directly change the CDM. Before the design enters into a new iteration, the earlier version of the design data which is in CDM is numbered and stored separately in order to maintain the data from all the design iterations. Since all the models use the parametric information from the CDM, any previous version of the design can be reproduced using the software API and the concerned data file. In general, a model can be defined as the abstract representation of a concept, a phenomenon or a physical entity, a data system can be used to describe and analyze the model totally if all the associated constraints are known.

Chapter 5
Knowledge Capturing Method

As discussed earlier, in order to create an efficient and time saving design process, it is required to automate the product development stages as much as possible. In order to fully utilize the potential of the modeling automation, the algorithm development process needs to be simplified so that is should be used during every design problem. The two data sets involved in development of computer based design process are: (1) automated knowledge areas specific to product as listed below, and (2) file handling and information flow mechanism. The automation of data and files handling is easy in a programming environment. Thus in order to achieve the automation of design process it is important to find a convenient methodology for automation of various knowledge areas. For a design process which involves limited number of iterations, creation of program for model generation consumes more resources and cost compared to manual process.

During product design, the initial stage of the design requires capturing the engineering design rules which are reflected by those related regulatory codes; standard component sizes have to be implemented into data structures as the captured knowledge in addition to following the commonly-accepted industrial design practice methodology. Once the program is developed, it is used to calculate all the required design parameter values automatically which are later used to create CAD and CAE models.

The calculated parameter values are then used in the generation of CAD and CAE models. In the modeling and analysis stage of the design proposed process, the knowledge embedded is in the form of standard modeling and analysis steps and the expert practice formulas associated with the process. Such knowledge is formally captured and used in the program code with the help of a kind of "journal file" created by applying a one-time GUI based interactive design and analysis procedure.

During the next stage of design, the process requires expert judgment and change management knowledge for design modification. For current study this is achieved by providing CAE analysis results directly to user and an interface is created to accommodate the intended design changes automatically by computer program embedded with engineering design formulae as mentioned above. In ideal scenario, in this stage the knowledge can be captured in the form of optimization algorithms and decision making matrices.

5.1. Knowledge Capturing by Using Journal File

One of the most difficult and time consuming task involved with automation of the design process is program code development for automatic generation of CAD and FE mesh model and CAE analysis. With increasing complexities in modelling and analysis, the API based program code becomes more and more lengthy and requires the user to have very detailed knowledge of model development and programming. Thus it is desirable to find a suitable and easy way to create program code for automatic product development process.

These obstacles can be overcome by using the journaling application in modern software tools coupled with parametric modeling. In current work, first a CAD model for the design object is developed interactively through the software tool GUI. At the same time the entire modeling is recorded in the form of program commands on the background through journaling which keeps the record of all the steps taken while model development with all the required of details. In software tools like NX6 the journal file developed can be embedded into a program code which develops the CAx model through API automatically.

5.2. Program Development for Generative CAD and CAE

In order to facilitate easy and rapid development of reusable algorithm, two functions of NX, i.e. journaling and parametric modelling, are used jointly with NX Open programming API. During the very first iteration of the design process, every product model and analysis is generated manually through the software GUI applications while recording every step taken by the designer using the journaling application. The journaling application for the software records every step in the form of required programming language in the form of algorithm functions in the same logic required to build the model and analysis. Every property of the model and analysis is imported from an external data repository which is the CDM for the design. Thus the journaling application provides a ready structure of programming functions involved with modelling process in required logic flow coupled with variable parametric modelling. Figure 5.1 shows the structure of the program development process for automatic generation of the computer models and analysis.

To fully utilize the software tool potential it is essential to predetermine the engineering modelling and design procedure development process. For a well defined process model, it is necessary to know the basic logic and algorithm required for constraint solving and management. In order to use journal file to create the program code to automatically generate all the computer models, the model needs to be generated using a specific set of steps during interactive modeling in software GUI. The first step in creating a journal file is assuring that all the previous data in the form of history is erased so that no previous data gets recorded into the journal. Before starting the interactive model development the designer has to make sure to have sorted the modeling process beforehand with the use of available functions compatible with journal recording application. Also the designer has to be familiar with the nature of the journal file to separate the portion useful to create the reusable program.

Journal provides a list of all the process flow and data in an orderly manner. By incorporating this data into a compilation structure along with organizing it with the help of logic, a programme code is developed with appropriate reusable data structure. Once the designer follows a specified set of steps to generate CAD model through software GUI a journal file in the form of a suitable programming language like C++ is recorded in the background which follows the exact steps and build-up logic as that used by the designer during interactive process.

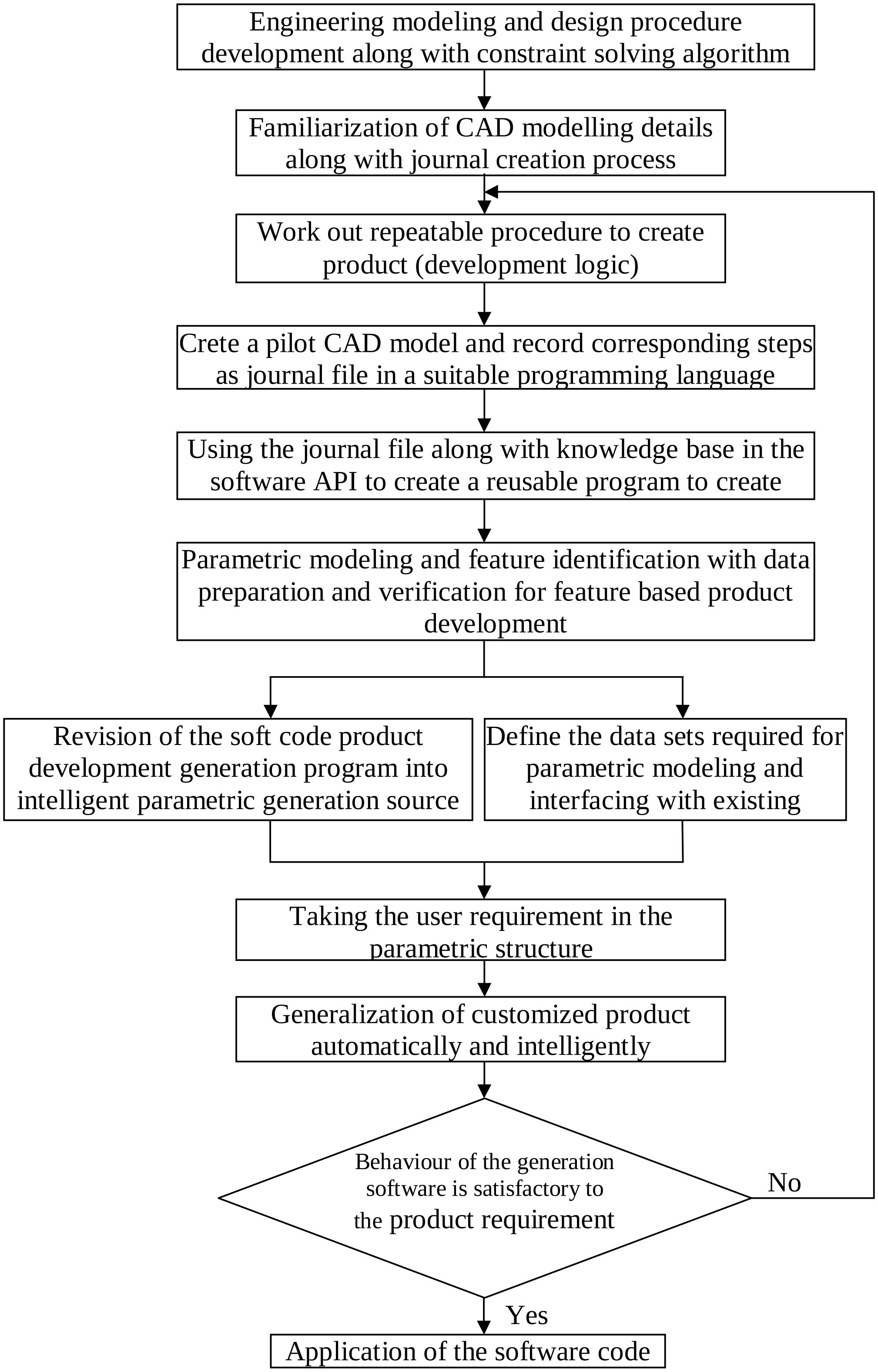


Figure 5.1: CAD and CAE program development process using a journal file

While developing the model through GUI, the designer has to make sure of associating all the feature related parametric information using an external data file (CDM) as input for parametric modelling. By associating journaling with parametric modeling, the program code generated becomes flexible following changes made into the external data file used as input. Each of the program code generated is the added to the proposed design process structure while using the knowledge embedded program code to manipulate the external data file creating a reusable and customizable product development structure.

By integrating the "journal" created through GUI into NX Open programming environment while associating each function with corresponding application, the journal can be used to develop the program which can be used to automatically generate the model. The application file is called into the program structure through individual header file specific to function used. This assures that the program follows the same logic as actual manual process and also makes sure that similar model development process is followed each time. Since the program exactly reflects the steps followed in GUI, any specialized steps required to create the model gets automatically embedded into it thus preserving the expert knowledge associated with the process while making it reusable. By capturing the designer's knowledge and intent associated with computer modeling and analysis, journaling enables generalization of customized product automatically and intelligently.

Although developing a program with the help of a journal file is fast and efficient, it has certain limitations. For NX package, all the software capabilities cannot be fully utilized with the help of journal application since it is not capable of recording every single manually operational function. Also while creating the journal it has to be assured that there is no retained history in the software as well as the system to assure a "fresh" start. Though the program code created is controllable and changeable up to a certain degree it lacks flexibility in case of restructuring for a completely different product. As the program code itself lacks capabilities for data and file handling, it has to be associated with an external data repository and file management system to make it flexible for changes. The program code generated using journal makes it difficult to re-edit the entire code or to rearrange the entire feature structure since everything is cramped into one single code with separating features into separate functions, thus the program code becomes rigid in the structure.

5.3. Conversion of Journal File Into a Reusable Program Code

The journal code provides an organized list of data types involved with model development. In order to use it in a reusable program code it is converted in to a data structure that can be compiled. Various changes required to convert a journal file to the reusable program code

5.2.1. Creating a Program Environment for Reusable Code

First step required during the conversion of journal into a program code is creating an algorithm environment which can be compiled as shown in Figure 5.2. To obtain the appropriate format for the program code, a program template provided by the NX open is used.

```
extern void main( char argc, char *argv[] )
{
   /* Initialize the API environment */
      if( UF_CALL(UF_initialize()) )
      {
            /* Failed to initialize */
            return;
      }
   /*Application code starts*/
      Session *theSession = Session::GetSession();
      ...
      ...
   /* Terminate the API environment */
   UF_CALL(UF_terminate());
}
```

Figure 5.2: Creating a program environment for reusable code

5.2.2. Replacing Hard Coded Functions With Variables

When a journal is created it records all the steps that the software takes while model development. During the process all the expressions and associated functions in the form of hard logic are also recorded as shown in Figure 5.3. By associating the functions with appropriate variables imported in the form of expressions.

Journal

```
/* Hard coded functions with variables*/
  expression2417->
  SetRightHandSide("15.25");
  expression2418->
  SetRightHandSide("26.00");
  expression2419->
  SetRightHandSide("-29.0");
  …

/* Perpetual trial and error based recorded functions*/
  Expression *expression2153;
  expression2153 = associativeArcBuilder22->Radius();
  expression2153->SetRightHandSide("94");
  …
  Expression *expression2170;
  expression2170 = associativeArcBuilder22->Radius();
  expression2170->SetRightHandSide("100.5");
  …
```

Reusable code

```
/* Hard coded functions with variables*/
  expression2417->SetRightHandSide
  ("Saddle_dimension_A/2");
  expression2418->SetRightHandSide
  ("Saddle_location_1-(Saddle_dimension_C/2)");
  expression2419->SetRightHandSide
  ("-Saddle_dimension_B");
  …

/* Perpetual trial and error based recorded functions*/
  Expression *expression2170;
  expression2170 = associativeArcBuilder22->Radius();
   workPart->Expressions()->EditWithUnits
  (expression2170, unit1,
  "Gas_nozzle_diameter+Gas_nozzle_thickness");
  …
```

Figure 5.3: Replacing hard coded functions with variable input

5.2.3. Deleting Un-necessary Data and Functions

As the journal file is developed during creating a model through GUI, it records un-necessary graphical information such as the modelling view information. Also it records intermediate and temporary functions and variables generated as shown in Figure 5.4. Since this information is specific to a single problem and is not required in the program code, it needs to be removed.

```
/* Function for modeling view orientation*/
 ModelingView *modelingView1;
 modelingView1 = workPart->ModelingViews()->WorkView();
 modelingView1->SetRenderingStyle
 (View::RenderingStyleTypeStaticWireframe);
 modelingView4 = workPart->ModelingViews()->WorkView();
 Point3d scaleAboutPoint1(39.8975, 21.9294, 0.0);
 Point3d viewCenter1(-39.8975, -21.9294, 0.0);
 modelingView1->ZoomAboutPoint(0.8, scaleAboutPoint1,
viewCenter1);
 …

/* Intermediate expressions*/
 workPart->Expressions()->SystemRename(expression84, "p25__x");
 workPart->Expressions()->SystemRename(expression85, "p26__y");
 workPart->Expressions()->SystemRename(expression86, "p27__z");
 …
```

Figure 5.4: Deleting un-necessary data and functions

5.2.4. Grouping Similar Sections and Functions Together

A journal records the model development process step by step thus it keeps the track of each individual process. Thus a journal file contains repetative functions and step by step incrementation of expression values. In order to convert journal into a reuable code, all the similar functions and sections are grouped together as shown in Figure 5.5. The incremental expressions are also replaced by variables.

Journal

```
Section *section14;
section14 = joinCurvesBuilder1->Section();
section14->SetDistanceTolerance(0.001);
...
Section *section15;
section15 = joinCurvesBuilder1->Section();
section15->SetChainingTolerance(0.00095);
...
Section *section16;
section16 = joinCurvesBuilder1->Section();
section16->SetAngleTolerance(0.5);
...
```

Reusable code

```
Section *section14;
section14 = joinCurvesBuilder1->Section();
section14->SetDistanceTolerance(0.001);
...
section14->SetChainingTolerance(0.00095);
...
section14->SetAngleTolerance(0.5);
...
```

Figure 5.5: Grouping data sections and functions together

5.2.5. Associate the Program Code With Appropriet “.dll” Files

When the journal file is recorded it includes all lists of all the related header files required. In order to compile the program code, the header files need precompiled .dll files to be included in the program debug directory.

5.2.6. Maintaining Associativity Between Design Parameters

Maintain the associativity between design features by developing a model with well defined and semantically inter-connected design parameters. The parameters are interconnected with engineering design concepts and constraints. With well defined parameters, constraints and interdependencies between various features is maintained.

Chapter 6

Application of Proposed Methodology

(Two Phase Separator Design)

6.1. Initial Product Development Stage

During the initial product development stage using engineering design formulae and industrial standards, all the required parameters associated with the separator are calculated. The standards used for design are ASME Boiler and pressure vessel code section 8. With the help of C++, above mentioned design process for the separator vessel design is stored in programmed code as embedded engineering knowledge. In a standard working scenario, the designer is provided with operating conditions for the separator vessel in the field. The first requirement for the user to select the type of vessel required, in this case it is either vertical or horizontal vessel. The input data includes requirements such as flow rate and fluid properties along with the working conditions such as pressure and temperature etc.

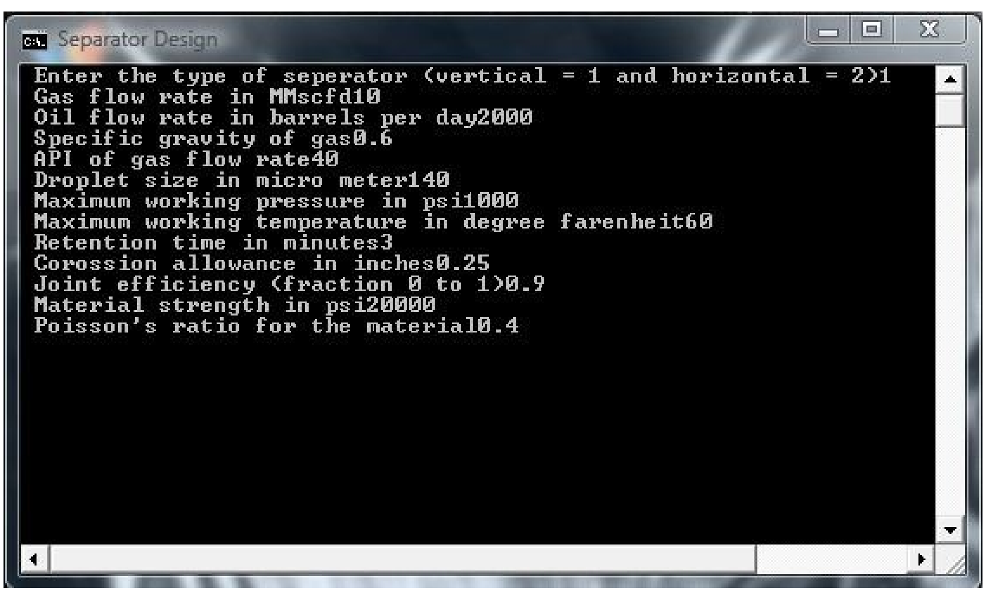


Figure 6.1: User input console

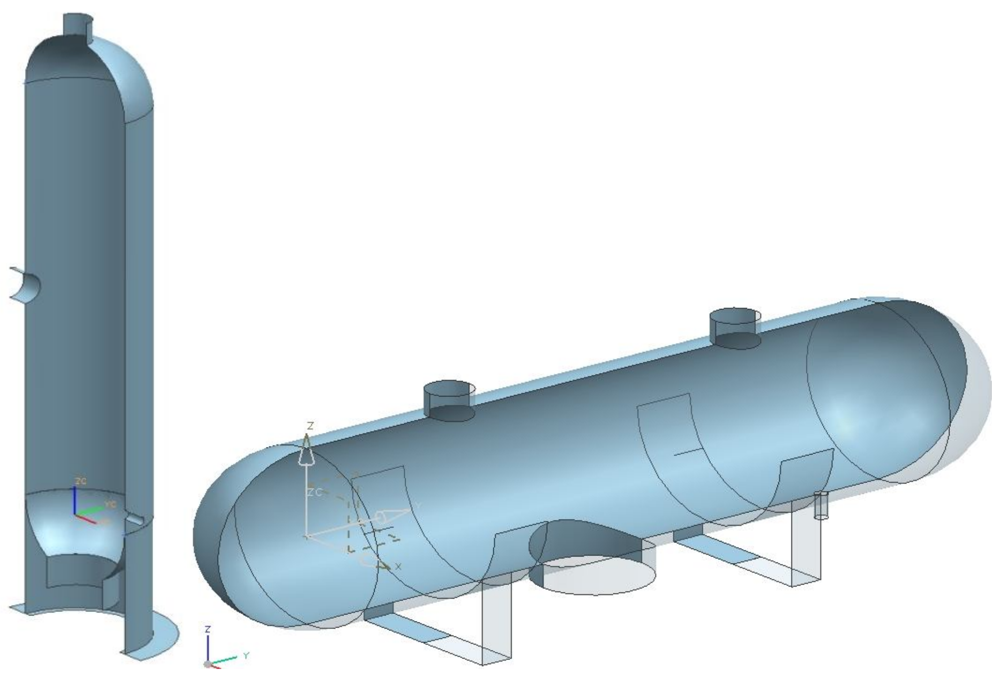
Flow of the program code is set up to follow the standard design procedure (figure 3.4) followed to solve an engineering problem. First all the input values are converted from field units to the unit category used for calculations; in this case all the calculations are done with BS units. Next all the relevant non geometric parameters associated with operation and flow are calculated, this includes gas flow rate, liquid flow rate, Maximum allowable working pressure, allowable stress values etc. This is the essential engineering information in order to calculate the geometric parameters required to build the CAD, FEM and CAE models for the required type of vessel. All the standard sizes for the components such as vessel diameter and nozzle sizes are stored in external files and can be modified as per the requirements by user. This provides a flexible option for user to design the vessel using the available components in inventory. The calculation steps for the vertical and horizontal separator vessels are the same; the difference is for the design of the support and support requirements. For vertical vessel, the skirt is designed using design equations coupled with standard sizes where as the dimensions for the horizontal vessel saddle are standard for given vessel diameter. Some of the assumptions made for the design process are

1. The vessel is design for the loading of internal pressure.
2. The vessel is designed with essential working components.
3. The pressure and temperature is assumed to be uniform
4. Design is only for static loading.

Once all the calculations are done, the resulting parameters are stored in a text file automatically using the C++ interface. This file then serves as common parametric input for all the design models hence forth.

6.2. Conceptual Design Stage

During the first phase of design which is the conceptual phase, the separator model only contains basic parameters. It is essentially a step to verify the design of the basic vessel of the separator. It also contains all the nozzle openings and connections and vessel support. Figure 6.2 shows the mid-plane conceptual model for vertical and horizontal separator. Conceptual model is mid- plane planar model essentially to check the plane stress and plane strain to validate vessel design. The mid plane CAD model serves as the basis to create the mid place FE model converting the planar model into 2D planar elements. The associated thickness of each design feature is added as a physical property of the associated mesh. The simulation is run for static load considering internal pressure load and temperature load. For simulation purpose, base of the vessel support is fixed along with the ends of nozzles to simulate contact with ground and pipe continuations respectively. Once the simulation is run the major values to check are maximum deformation and maximum stress.



(a) Vertical separator **(a) Horizontal separator**

Figure 6.2: Mid plane CAD models

Figure 6.3 shows the simulation results for vertical and horizontal separators. If the results are acceptable as per the designer's requirements, the design is passed on to further stage otherwise the user is prompted through program API to make design changes. Changes made are further verified again by program code to make sure that they follow the standard design procedure and codes. Once verified, the refined parameters in the CDM are again used to conceptual design is finalized, the design process proceeds to the next phase. Even after this phase if during further refinement basic vessel design needs correction, it is possible to get back to this initial iteration as shown in the process diagram in Figure 3.4. Considering that all the models in this phase are planar and FE model contains 2D elements, time required to build all the models and the CAE analysis are very less compared to that of detailed 3D model. Also as most of the refinement of major design objects takes place during this phase, it reduces the number of iterations required during the detailed design phase thus reducing the overall time required.

All the above mentioned process takes place automatically through NX Open external application file which is coupled with basic design process program code structure. The thickness associated with each design feature is assigned during FE stage as physical property.

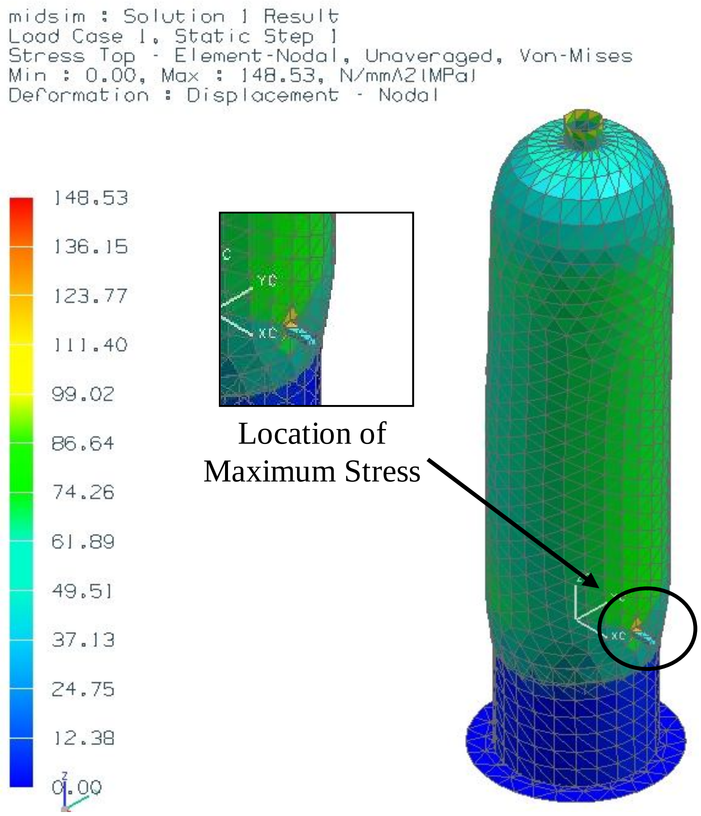


Figure 6.3: Von-Mises stress (MPa) analysis result generated from NX Nastran for parametric mid-plane model of vertical separator

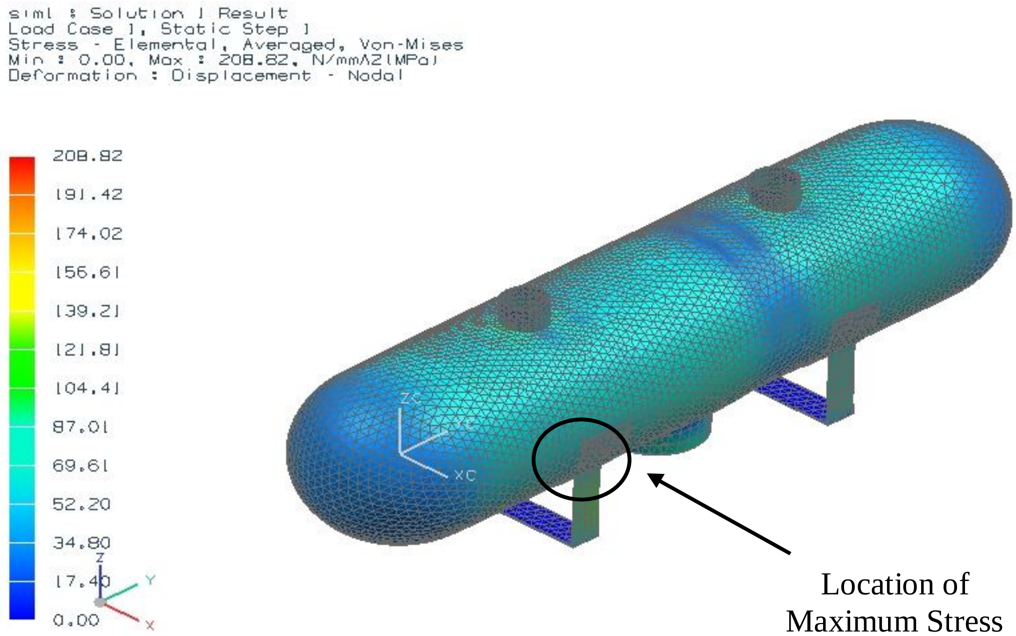
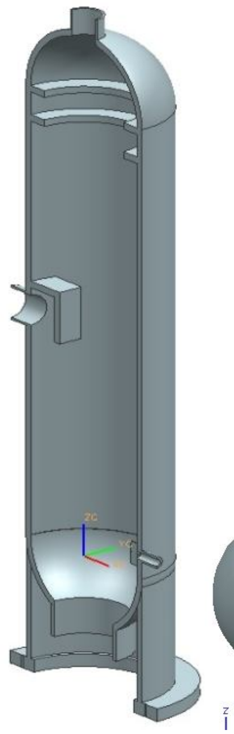


Figure 6.4: Von-Mises stress (MPa) analysis result generated from NX Nastran for parametric mid-plane model of horizontal separator

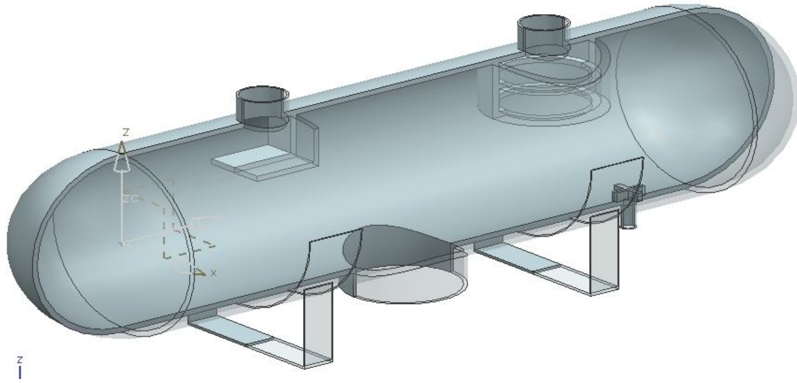
During mid-plane design, major objective is to make sure that the base vessel of the separator is acceptable as per user along with the structural support without considering internal details. Simulation run time for the mid-plane model of the vessel is approximately 1/3 that of the simulation run time for equivalent 3D solid model. Also mid-plane model without any major details provides location of maximum stress and deformation for the basic vessel structure thus making it convenient make changes to it. This initial analysis also provides stress distribution at major nozzle openings and at the supports. Also with the absence of thickness feature and minute details in the FE model, a higher mesh size can be used to represent the model without losing any of the analysis accuracy.

6.3. Detailed Design Stage

During second phase of design process detailed design of separator is produced. Figure 6.4 shows detailed 3D models for vertical and horizontal separators. This model is 3D solid model with full design details. Separator model includes internal details such as internal diverter, vortex breaker and mist extractor support. Just like the mid-plane model, solid 3D models are also generated automatically using NX6 API called NX Open.



(a) Vertical separator



(a) Horizontal separator

Figure 6.5: 3D Solid CAD models

3D CAD model is used to create the FE model required to create the simulation model and to perform analysis. This simulation is done with all the internal details, thus is much more detailed as compared to the analysis of the mid-plane analysis. Simulation conditions for solid model analysis are same as mid-plane analysis. All the constraints and loads are also similar. Just like during mid-plane phase the model is refined until desired analysis results are obtained. Parameters checked for design verification include maximum stress and maximum displacement. Figure 6.5 shows the simulation results for detailed models of vertical and horizontal separators. This is the second iteration in the design process. The versioned solid CAD model is separator structures with all basic essential functional components. The model then applied with required tolerances and process details can be used for manufacturing. The CAD model can then further be used for CAM process. As per the conditional requirements, while manufacturing additional components may need to be added later such as de-foaming plates and flow breakers.

Since 3D model has to be meshed to accommodate the thickness of vessel which is a considerably small dimension, the mesh size for the model is much smaller than that used for equivalent mid-plane model. Thus because of increased number of components, 3D mesh and decreased mesh size, simulation takes considerably more time compared to that with mid-plane model. FE analysis with 3D model also yields maximum stress and deformation values and their corresponding locations. But in this analysis, the model has a lot more contact areas and stress concentration areas compared to mid-plane model. Hence with this simulation, the emphasis is to make sure that the results associated with detailed assembly are acceptable and within limits. Location of maximum stress and deformation may vary in 3d and mid-plane models because of stress concentration and added stiffness because of internal details.

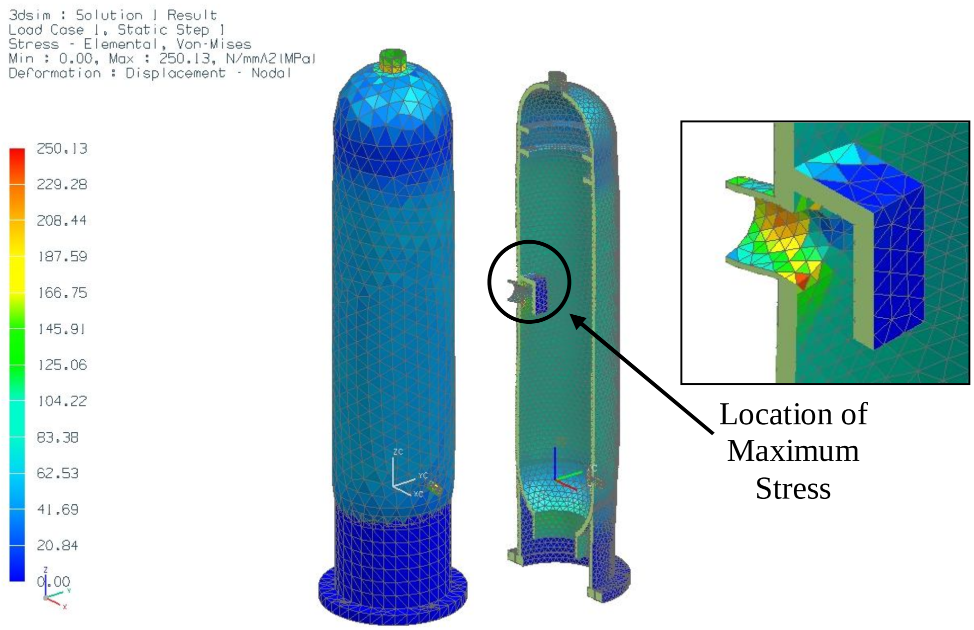


Figure 6.6: Von-Mises stress (MPa) analysis from NX Nastran for parametric 3D model of vertical separator

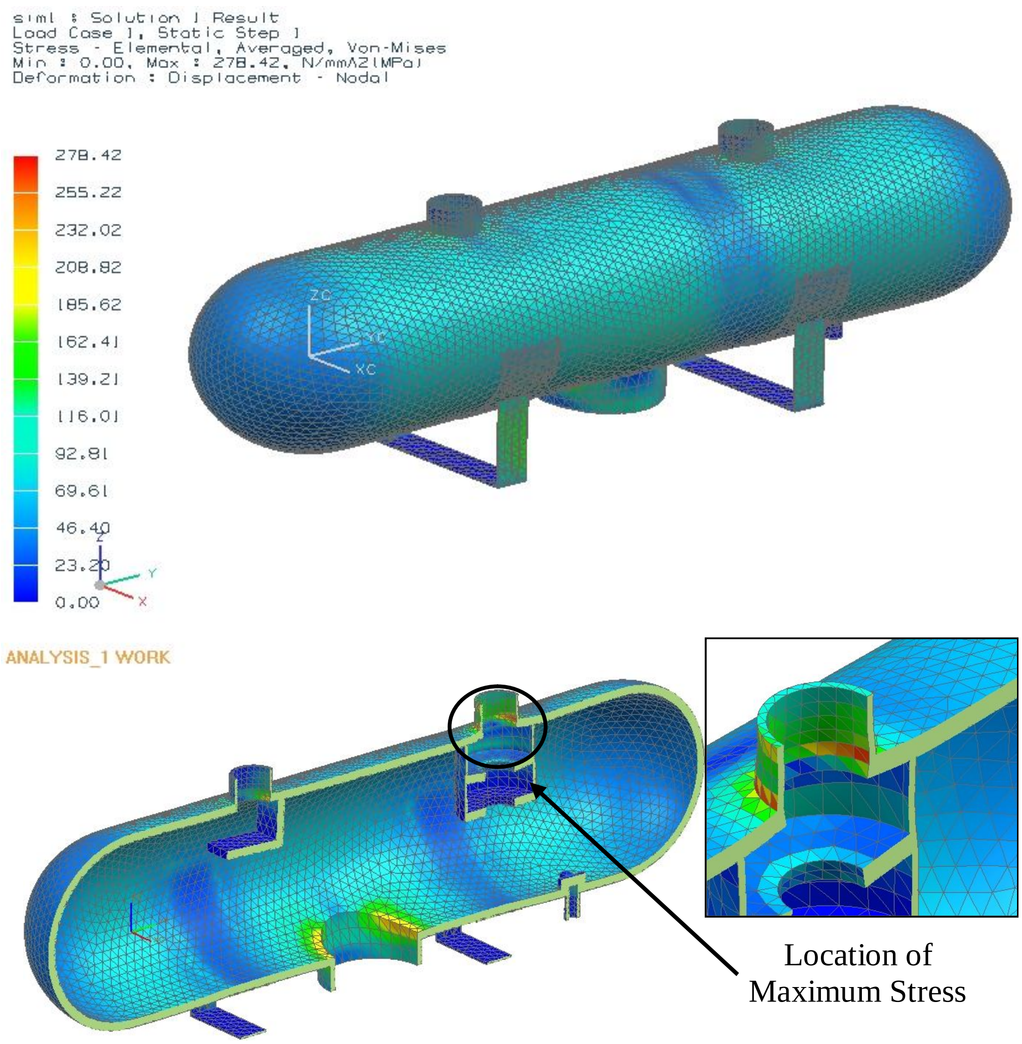


Figure 6.7: Von-Mises stress (MPa) analysis from NX Nastran for parametric 3D model of horizontal separator

6.4. Variation Change Scenarios

After every design iteration stage, the user is provided with the simulation results. After manual inspection of the analysis results in NX Nastran post processor with respect to desired requirement, user needs to make a decision on whether the design model created is acceptable or need further refinement. FE analysis of each model as mentioned earlier provides maximum structural static stress and deformation and their respective locations. Thus major requirement is for these parameters to be within the acceptable range. If the user decides to make changes to the design model, user is provided with options to modify the design further.

The user is provided with following modification options

1. Modification of the operating conditions of the vessel
2. Modification of the material properties
3. Direct modification to the automatically created data file to change parameters as per requirement

Once the user does the required modifications, the program recalculates all the parameters and makes the appropriate changes to the parametric data model and the design models are recreated automatically using the software API. Modification to operating parameters changes the scope and general requirements of the design and thus the process starts from the beginning all over again. By changing the material properties, the design still retails its operational intent and most of the major dimensions but with better material, the vessel can withstand applied load better and gives a relatively safer design. But both of the above options require re-running the program code in order to recalculate all the parameters again since all the parameters in CDM are semantically connected to each other. An expert designer may want to change a single particular parameter, like the shell thickness, without making any other changes to design, thus in order to do that, the only way is to directly manipulate CDM and run the program code to again automatically generate all the design models again. Thus this method offers a great deal of flexibility in order to incorporate changes into the design.

6.5. Finite Element Analysis Details

The computational analysis method used for CAE for the current study is the Finite Element Method. Finite element method is based on dividing of the design model into smaller elements and calculating the loads associated separately for each element and then combining the results for the entire model to get final results. Design of the separator vessel was done under the loading of internal pressure and temperature. All the loads on the structure are static and the material is assumed to be homogeneous and isotropic.

First step of the analysis requires meshing the design CAD model available in suitable mesh elements. One of the important task while mesh building is to select an optimum mesh size. The mesh element size has to be small enough to produce a reliable results but it should be big enough so that it will take minimum computational time. For any given vessel design, the smallest feature of the model are the various thickness associated with nozzles and vessel body. In case of mid-plane model mesh, each part of the model is meshed separately to assign specific mesh and physical properties.

With mid-plane mesh model, as separate parts of the design object has no physical thickness associated in the form of geometry, the thickness has to be added as a parametric physical property. Separate components in the mid-plane model are then connected to each other using one dimensional connecting elements to maintain the associativity. The connecting mesh elements are mass-less and rigid thus only serves the purpose of transmitting the loads and does not affect the structural integrity of the model. Contact between the vessel body and the vessel support is illustrated with the help of 0ne dimensional bar elements to establish a surface to surface interaction. In 3D solid model, mesh size was chosen to get uniform meshing for these small features which would ensure that it is compliant with all the other larger features. In mesh modeling process for current study, the mesh creation is automatic by using NX Open coupled with parametric modeling where as various physical and material properties needs to be assigned manually. The element type used for meshing is tetrahedral to get a uniform distribution around openings and corners as well as to get maximum available nodes.

The next stage of analysis process is using the mesh model to apply specific loads and constraints at required locations automatically by NX Open and parametric modeling using CDM. Figure 6.6 shows the constraint and load map for a horizontal separator analysis model.

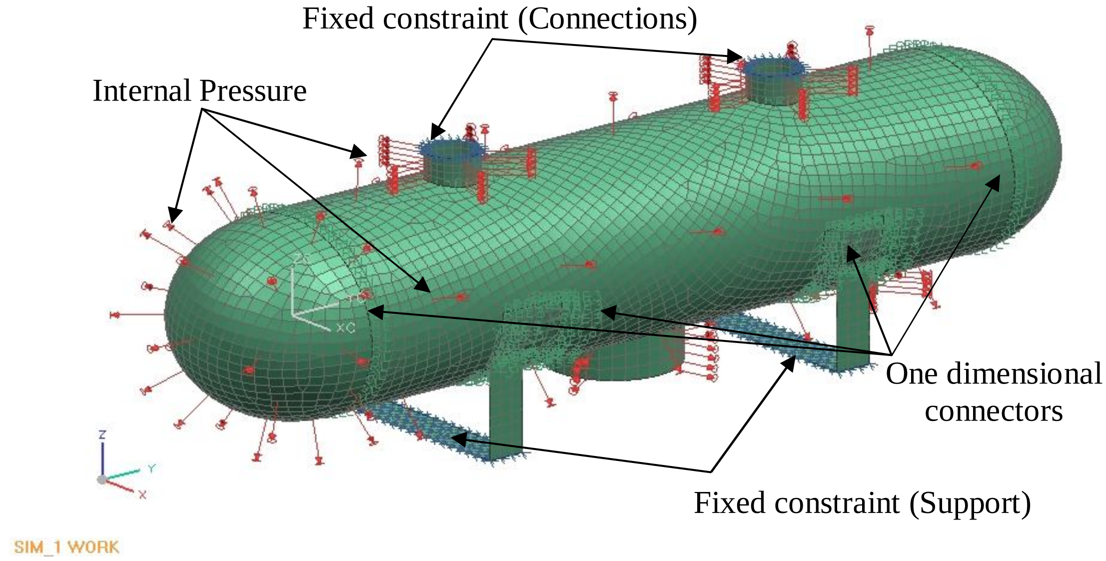


Figure 6.8: Load and constraint model

Internal pressure is applied on the main vessel body as uniformly distributed load directed outwards from the vessel surface. The temperature is also applied to the vessel body and is assumed to be uniformly distributed on the entire surface. Vessel support base at the ground end is given fixed constraint to represent fastening. Whereas the nozzle ends are constrained for axial movement along the pipe continuation in respective dimensions. The model then goes through static analysis calculations and results in the form of maximum Von-Mises stress and maximum deformation are obtained. Since the software unit system is set to SI standards, the stress values are obtained in MPa where as deformation values are in Millimetres. Once the designer is provided with analysis and observes the results in post processor depending upon the constraints requirements, additional analysis conditions can be added to re-evaluate the solution. In case of multiple loads and constraints, the iterative solver should be used to get a more refined and accurate solution.

6.6. Example Case Results

For current study, a design problem of moderate pressure and temperature application separator is used from [54]. Similar operational requirements are used for the design of both horizontal and vertical separator case. All the design parameters are calculated automatically according to the above mentioned design process while using the standard component size in Appendix B. Table 6.1 shows all the initial requirements or “the inputs” used for the design.

Table 6.1: Input for the separator design case study

CDM parameter name	Engineering parameter symbol	Definition	Unit	Value
R1	*dm*	Droplet size in fluid to be separated	µm	140
R2	*Ql*	Liquid flow capacity	BPD	2000
R3	*Qg*	Gas flow capacity	MMscfd	10
R4	*tr*	Retention time	min	3
S1	*P*	Operating pressure	psi	1000
S2	*T*	Operating temperature	°F	60
S3	*API°*	Specific gravity of oil	-	40
S4	*SGg*	Specific gravity of gas	-	0.6
S5	*Ss*	Material Strength	psi	20000
S6	*E*	Joint efficiency	-	0.9

$\mathbf{R_n}$**:** Customer operational requirement parameters;

$\mathbf{S_n}$: Technical Specification parameters

The initial values of some of the major design parameters calculated by the analytical method during the first run are as below in Table 6.2 along with the results for the first simulation run.

Table 6.2: Design Results

Engineering parameter symbol	Definition	Unit	Value
D	Shell diameter	in	36
Lss	Shell length	in	120
Hh	Head height	in	18
dng	Gas Nozzle diameter	in	8
ts	Shell thickness	in	1.755
tng	Gas nozzle thickness	in	0.625
S	Maximum stress	psi	34000

Though the design is done according to analytical method but the FE analysis in figure 6.7 the maximum stress value (34969 psi/248 MPa) is much higher than the allowable stress values (20000 psi). There are several methods which can be used to improve the design. By increasing the thickness of the vessel from 1.755 to 2.1 the results obtained are within acceptable limit. Also the designer may choose to use a higher grade material or to add some additional reinforcements near the stress concentration area. Depending upon several conditions such as the cost associated with each option along with the use of extra material and manufacturing efforts the correct option is chosen by the user.

Chapter 7

Advantages and Limitations

The proposed design method is a combination of conventional design process with CAD-CAE integration for better efficiency and for partial automation. With the help of modeling and analysis software API programming, the development time and efforts associated modeling are greatly reduced. By using the journaling application for program development, it reduces the need for programming expertise as well as development time associated with it. It also provides flexibility to handle complex modeling and analysis problems. With the help of journaling it is easier to capture modeling and analysis expertise and making it reusable.

Since the process makes integrated use of the CAD and CAE software capabilities it helps to reduce the model and analysis time along with program development time while keeping with the traditional design approach. Use of mid-plane analysis reduces the time associated with the product development process during initial stages. Since information associated with design is stored in a neutral external data file in the form of parameters and used as input for modeling and analysis, CAD/CAE integration is more versatile and flexible. Also it makes it easier to transfer and control the specific information associated with different models as well as coupling the computer modeling process with engineering knowledge embedded program. Parameter based modeling and analysis makes it easier to focus changes associated with specific design features.

The CAD and CAE integration process with CDM offers centralized design parameters and their data and yet the ease of separate automation for CAD and CAE. It facilitates the manipulation of design data and the control over parameters with the use of engineering knowledge along with CAD and CAE for design cycles. This approach also incorporates use of rules and standards checking in a separate module reducing the dependencies among different sub-domain models which have the generic applications as CDM-based programming allow flexibility of adopting different practice standards used for design. CDM manipulates data on parametric level thus design changes can be successfully propagated to all the related features of the CAD and CAE models by regeneration which in turn eliminates requirement of specific feature editing and manipulation. The parametric data model developed includes manufacturing consideration on design such as the welding efficiency. CDM data can be further customized with the help of well-organized standard component dimensions and configurations adapted to the proposed process; then the design features can consider the available inventory along with the required installation constraints.

The proposed integration method though quiet flexible, still has some limitations, such as the need for the initial CDM parameter and relation identification beforehand to develop the model generation program of the process management module along with the required standards and governing codes. Thus the initial development phase requires considerable programming. Also the design procedure needs to be sorted beforehand along with the required assumptions to build CAD model and CAE analysis and associated constraints to develop the logics required for automatic generation. Thus the process only offers long term efficiency for well established, generic and set design problems. However, for those cases without established design procedures and with a lot of "ad-hoc" user interventions and "rolling" backward or forward, this method seems lack of the flexibility in comparison to interactive modeling and analysis approach.

The process has limited capabilities of programming with the help of journal file and workability of the process with different software tools. Also it requires initial training to understand software API programming. To create a workable journal file as well as to develop an efficient program to generate computer models, it is essential to predetermine all the steps involved with modeling and analysis process. Use of journaling provides less control over the program development and modification process. Also the process requires capturing the journal file with the same programming base as that of the NX Open API to accommodate journal record into code.

Chapter8

Conclusions and Future Work

The major objective of the research is to make the overall design development process faster and more flexible to cope up with rapid changes. This work tried the integration of modern CAD and CAE software tools. A generative, continuous and evolving design and analysis process is proposed. The proposed process has the capability to handle CDA and CAE integration with more efficiency. Automation of file and data management has been used to improve the efficiency of the overall process. A reusable integrated data file, i.e. the CDM, is designed to centralize the parameters involved throughout the design evolvement processes across different design software applications, including CAD and CAE processes. In addition, a reusable program has been developed in which design rules that are specific to the related engineering principles, constrains, and expert knowledge are embedded. A generative and parametric modeling approach was applied. Thus, when changing design contents, a semi-automatic design process involving minimum human interactions is developed. The use of CDM, a common parametric data model to semantically connect all design models, represents an innovative idea of this work, enables the designer to have a better control over the process.

The other innovative idea is the use of journal file for the extraction of design knowledge and procedures used for automation of computer generated models through APIs. The adaptive nature of the “learning” process via journal files enables faster development of product generation codes with similar development process. With the use of external executable files and common file formats for CAD and CAE modeling processes, the proposed methodology offers flexibility of using different CAD and CAE software tools.

Integration of CAD and CAE at parametric level using a CDM enables to solve the problem of association of feature-based semantic knowledge and the iterations of CAD and CAE interaction cycles. With the help of CDM, it is feasible to integrate design and analysis processes via the associative relations and the built-in interfaces with the CAD and CAE models. With this generative approach of design, design cycles can be coherently modelled with a systematic updating mechanism with reusability of engineering knowledge and the design expertise. With a well managed program design structure, the method is not limited by the software tools used. With a neutral data structure, the CDM gives the flexibility of using various CAD and CAE software tools; their APIs and can be used to automate the entire modeling process. Thus, this approach potentially saves a significant amount of time associated with the design process.

CDM via programmed design management structure connects the design models and expert knowledge with any KBE implementation; it separates the programming of design expertise from CAD or CAE modeling hence ensures the reusability of those design procedures once they are created in a computerised design format. All the design and analysis information is stored independent of actual CAD or CAE software tool thus any loss of information during the model translation or derivation can be retrieved from the CDM. CDM also acts as automatic input through API of modelling software tool with the help of parametric modelling capability, thus eliminates need for direct human interaction in the process. This minimizes possibilities of error as well as ensures that the same process flow is followed through different iterations. The complexity, accuracy and quality of design depends upon the embedded knowledge and expertise as well as creating an API template for CAD and CAE, a well developed system with a CDM can efficiently handle complex design problems.

Current work with CDM is limited with only CAD and CAE but the use can be further explored in the fields of CAM and also for other uses such as cost estimation. Future enhancement in the work involves creating a knowledge based software tool for automatic assembly coupled with part template library to deal with more diverse design problems. This study only involves the preliminary design of a pressure vessel type limited to sizing and the essential operational concepts; further work will involve refining the model with more design features to prepare a complete drawing ready for manufacturing.

The process can be further improved by including more automation into the process along with refining the data and file management structure further coupled with optimization algorithm for decision making. With the help of a well defined database for standard components as well as a library for modeling API program codes for various products, the process can be made to automatically adapt to requirements. As mid-plane model approach is limited to uniform thickness products, a further simplification technique needs to be developed for more complex products.

List of References

[1] Radhakrishnan, P., Subramanyan, S., Raju, V., CAD/CAM/CIM, New Age International, Daryaganj, Delhi, IND, 2008.

[2] J. J. Shah, M. M. Ntylä, Parametric and Feature-Based CAD/CAM: Concepts, Techniques, and Applications, Wiley-interscience, 1995.

[3] Pahl G, Beitz W, Feldhusen J, Grote K-H, Engineering Design—A Systematic Approach. Wallace K, Blessing L (Trans. and Eds.) 3rd ed., Springer, Berlin, 2007

[4] T. Tomiyama , P. Gu, Y. Jin, D. Lutters, Ch. Kind, F. Kimura, Design methodologies: Industrial and educational applications, CIRP Annals - Manufacturing Technology, v 58, n 2, p 543-565, 2009

[5] R. Anderl And R. Mendgen, Parametric Design And Its Impact On Solid Modeling Applications, Symposium on Solid Modeling and Applications, p 1-12, 1995

[6] L. Susca, F. Mandorli, C. Rizzi And U. Cugini, Racing car design using knowledge aided engineering, Artificial Intelligence for Engineering Design, Analysis and Manufacturing, v 14, n 3, p 235-49, June 2000

[7] C.M. Hoffmann, K.-J. Kim, Towards valid parametric CAD models, CAD Computer Aided Design, v 33, n 1, p 81-90, January 2001

[8] Zhu Y., Yin M., Fang D., Research on parametric design approach of body skeleton of passenger car, in proc. Of International Technology and Innovation Conference 2006, ITIC 2006, IET Conference Publications, n 524, p 1527-1531, 2006,

[9] J. Monedero, Parametric design: a review and some experiences, Automation in Construction, v 9, n 4, p 369-77, July 2000

[10] S. Myung and S. Han, Knowledge Based parametric design of Mechanical products based on configuration design method, Expert Systems with Applications, v 21, n 2, p 99-107, August 2001

[11] Zuo B. C. And Li D X., An Object Oriented intelligent CAD system for ceramic kiln, Knowledge-Based Systems, v 14, n 5-6, p 263-70, July 2001

[12] M. Hauser and R. J. Scherer, Application of intelligent CAD paradigms to preliminary structural design, Artificial Intelligence in Engineering, v 11, n 3, p 217-29, July 1997

[13] Henrik K. J., Some general aspects of FEM usage, International Journal of Computer Applications in Technology, v 7, n 3-6, p 124-130, 1994

[14] P. Wriggers, M. Siplivaya, I. Joukova and R. Slivin, Intelligent support of engineering analysis using ontology and case-based reasoning, Engineering Applications of Artificial Intelligence, v 20, n 5, p 709-720, Aug. 2007

[15] M. Pinfold, C. Chapman, The application of KBE techniques to the FE model creation of an automotive body structure, Computers in Industry, v 44, n 1, p 1-10, January 2001

[16] M. Novak and B. Dolsak, Intelligent FEA-based design improvement, Engineering Applications of Artificial Intelligence, v 21, n 8, p 1239– 1254, Dec. 2008

[17] Q. Li and W.J. Zhang, Application of model-based reasoning to the development of intelligent CAE systems, Engineering Applications of Artificial Intelligence, v 11, n 3, p 327-336, Jun 1998

[18] D.C.Y. Yip, M.C. Law, K.P. Cheng, K.H. Lau and S. Barnes, Knowledge intensive CAD in product design validation, International Journal of Knowledge-based and Intelligent Engineering Systems, v 9, n 2, p 45–61, Nov. 2005

[19] C. J. Anumba, Functional integration in CAD systems, Advances in engineering software, v 25, n 2-3, p 103-109, Mar-Apr 1996

[20] M.S. Shephard, M.W. Beall, R.M. O'Bara and B.E. Webster, Toward simulation-based design, Finite Elements in Analysis and Design, v 40, n 12, p 1575–1598, July 2004

[21] Schreier P, Modelling: CAD/CAE: Closing the CAD/CAE gap, Scientific Computing World, n 109, p 4 , Dec. 2009-Jan. 2010

[22] A. Van der Velden, CAD to CAE process automation through iSIGHT-FD, In proc. of the ASME Turbo Expo, v 1, p 87-93, 2007

[23] G. Foucault, J.-C. Cuillière, V. Francois, J.-C. Lèon and R. Maranzana, Adaptation of CAD model topology for finite element analysis, Computer-Aided Design, v 40, n 2, p 176–196, Feb. 2008

[24] B. Xu, N. Chen, An integrated method of CAD, CAE and multi-objective optimization, In the proc. of the 2009 IEEE 10th International Conference on Computer-Aided Industrial Design & Conceptual Design (CAID&CD 2009), p 1010-14, 2009

[25] Wei S., Q. Ma and S. Chen, A framework for automated finite element analysis with an ontology-based approach, Journal of Mechanical Science and Technology, v 23, n 12, p 3209-3220, January 2010

[26] E. Aziz and C. Chassapis, Knowledge-based geometry generation for spur and helical gears, Concurrent Engineering: Research and Applications, v 10, n 3, p 251-61, Sept. 2002

[27] C.B Chapman and M. Pinfold, Design engineering—a need to rethink the solution using knowledge based engineering, Knowledge-Based Systems, v 12, n 5-6, p 257-267, October 1999

[28] G. Colombo, A. Mosca, F. Sartori, Towards the design of intelligent CAD systems: An ontological approach, Advanced Engineering Informatics, v 21, n 2, p 153-68, April 2007

[29] X.Y. Xu, Y.Y. Wang, Multi-model technology and its application in the integration of CAD/CAM/CAE, Journal of Materials Processing Technology, v 129, n 1-3, p 563-7, 11 Oct. 2002.

[30] Y. Zhengfeng, J. Zhengfeng, Research on the integrated design of dual mass flywheel CAD/CAE/CAM, Key Engineering Materials, v 407-408, p 169-73, 2009

[31] Albers A, Leon-Rovira N., Aguayo, H., Maier, T., Development of an engine crankshaft in a framework of computer-aided innovation, Computers in Industry, v 60, n 8, p 604-12, Oct. 2009

[32] B.-W. Cao, J.-J. Chen, Z.-G. Huang and Y. Zheng, CAD/CAE integration framework with layered software architecture, in proc. of 11th IEEE International Conference on Computer-Aided Design and Computer Graphics, Huangshan, China, 2009, p 410-415

[33] J.A. Penoyer, G. Burnett, D.J. Fawcett, S.-Y. Liou, Knowledge based product life cycle systems: principles of integration of KBE and C3P, Computer-Aided Design, v 32, p 311–320, 2000

[34] O. Hamri and L.-C. Lèon, Interoperability between CAD and CAE models for cooperative design, Methods and Tools for Co-operative and Integrated Design, p 451-462, 2004

[35] S. Arabshahi, D. C. Barton and N. K. Shaw, Steps Towards CAD-FEA Integration, Engineering with Computers, v 9, n 1, p 17-26, 1993.

[36] J. Johansson, Manufacturability analysis using integrated KBE, CAD and FEM, in the proc. of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, DETC 2008, v 5, p 191-200, 2009

[37] D.Z. Su and M. Wakelam, Intelligent hybrid system for integration in design and manufacture, Journal of Materials Processing Technology, v 76, n 1-3, p 23-28, Apr. 1998

[38] W F Resh, Computer aided engineering in engine design, International Journal of Vehicle Design, v 23, n 1, p 30-37, 2000

[39] P. Kagan, A. Fischer and P. Z. Bar-Yoseph, Integrated Mechanically Based CAE System, in proc. of the Symposium on Solid Modeling and Applications, p 23-30, 1999

[40] S. Zeng, E. J. Kim, G. M. Mocko, A. Xiao, R. Peak, F. Mistree, Systematic Design Method For Information Modeling In CAD/CAE, in proc. of the ASME Design Engineering Technical Conference, v 1 B, p 657-668, 2003

[41] X. Xu, U. Weiss and G. Gao, The Integration of CAD/CAM/CAE Based On Multi Model Technology In The Development Of Cylinder Head, International Journal of Automotive Technology, v 3, n 2, p 47-52, 2002

[42] G. Chen, Y.-S. Ma, G. Thimm and S.-H. Tang, Unified feature modeling scheme for the integration of CAD and CAx, Computer-Aided Design and Applications, v 1, n 1-4, p 595-601, 2004

[43] M. S. Smit and W. F. Bronsvoort, Integration of Design and Analysis Models, Computer-Aided Design and Applications, v 6, n 6, p 795-808, 2009

[44] Javier Monedero, Parametric design: a review and some experiences, Automation in Construction, v 9, n 4, p 369-77, July 2000

[45] Y.-M. Deng, G.A. Britton, Y.C. Lam, S.B. Tor and Y.-S. Ma, A feature-based CAD-CAE integration model for injection molded product design, International Journal of Production Research, v 40, n 15, p 3737-3750, Oct 2002

[46] Y. Kao, H. Cheng, C. She, Development of an integrated CAD/CAE/CAM system on taper-tipped thread-rolling die-plates, Journal of Materials Processing Technology, v 177, n 1-3, p 98-103, 3 July 2006

[47] G. Chen, Y.-S. Ma, G. Thimm and S.-H. Tang, Unified feature based integration of design and process planning, in the proc. of the International MATADOR conference, Manchester, United Kingdom, July 7-9, 2004, Srichand Hinduja (Ed.), Springer, p63-68, 2004

[48] G. Chen, Y.-S. Ma, G. Thimm and S. -H. Tang, Unified feature modeling scheme for the integration of CAD and CAx, Comput.-Aided Des. and Appl., v 1, n 4, p 595-601, 2004

[49] R.S. Peak, Characterizing fine-grained associativity gaps: a preliminary study of CAD-CAE model interoperability, in Proc. of the ASME Design Engineering Technical Conference, v 1A, p 573-580, 2003

[50] S. Zeng, R.S. Peak, A. Xiao and S. Sitaraman, ZAP: a knowledge-based FEA modeling method for highly coupled variable topology multi-body problems, Engineering with Computers, v 24, n 4, p 359-381, Oct. 2008

[51] S. H. Lee, A CAD–CAE integration approach using feature-based multi-resolution and multi-abstraction modeling techniques, Computer-Aided Design, v 37, n 9, p 941-55, Aug. 2005

[52] A. Mital, A. Desai, A. Subramanian and A. Mital, Product Development A Structured Approach to Consumer Product Development, Design and Manufacture, Elsevier Inc., 2008

[53] G. P. Gujarathi, Y- S Ma, Generative CAD and CAE Integration Using Common Data Model, in proc of The 6th annual IEEE Conf. on Autom. Sci. and Eng., Toronto, Canada, p 586-591, Aug 2010

[54] Maurice Stewart and Ken Arnold, Surface Production Operations Volume 1 Gas-Liquid and Liquid-Liquid Separators, Gulf Professional Publishing, 2009

Appendix A
Two Phase Oil-Gas Separator

Oil and gas separators are used to mechanically separate liquid and gas contents from a hydrocarbon extract of a reservoir at a specific pressure and temperature. All the necessary operating information as well as the design details for separator is obtained from Surface Production Operations Volume 1 Gas-Liquid and Liquid-Liquid Separators [54]. Design also follows the ASME pressure vessel design code. Separation is normally the initial phase of process facility, thus design of separator vessel is very important since a flawed separator can decrease the performance of entire plant.

The fluid from petroleum well can contain:

- Gas
- Condensable liquid vapours
- Water
- Water vapour
- Crude oil
- Solid debris

Phase separator is essentially a pressure vessel designed to separate a mixed stream of liquid and gas into relatively separate phases. Many of the downstream processing equipments in a facility can’t handle gas–liquid mixtures, for example:

- Pumps require gas-free liquid
- Compressor and dehydration equipment require liquid-free gas
- Product specification set limits on impurities
- Oil generally cannot contain more than 1% BS&W
- Gas sales contracts generally require that the gas contain no free liquids;
- Measurement devices for gases or liquids are highly inaccurate when another phase is present.

A.1. Separation Process

Reservoirs are usually at a much higher pressure then atmosphere, thus pressure of the fluid decreases as it rises thru the well. Therefore the capacity of the fluid to hold the dissolved gases decreases. The gas released from the fluid is held by the surface tension of the oil. When the fluid is warmed, this surface tension decreases and then gravity alone is sufficient to settle the heavy component in it. The important factors affecting the gravity separation are pressure, temperature, density of the fluid as well as the composition of the fluid. Thus a back pressure valves is generally used in separators to regulate the fluid pressure. Fluid temperature is maintained by heating or cooling the fluid by an external source or by expanding the fluid through a choke. Design and type of separator usually depends upon the composition of the fluid to be separated and thus it can be a two phase or a three phase separator with vertical or horizontal setup.

A.1.1. Factors Affecting Separation

The major factors affecting the design and operation of a separator are

- Gas and liquid flow rates (minimum, average, and peak)
- Operating and design pressures and temperatures,
- Surging or slugging tendencies of the feed streams,
- Physical properties of the fluids such as density and compressibility factor,
- Designed degree of separation (e.g., removing 100% of particles greater than 10 mm),
- Presence of impurities (paraffin, sand, scale, etc.),
- Foaming tendencies of the crude oil, and
- Corrosive tendencies of the liquids or gas.

The preliminary method of separation used is gravity separation, but along with that depending upon requirement in downstream processes, additional methods such as thermal separation, electrostatic precipitation, adhesive separation, adsorption separation etc are used.

A.1.2. Functional Sections of Gas- Liquid Separator

Each gas–liquid separator contains four major sections.

1) Inlet diverter

Inlet diverter is used to abruptly change the flow direction. It absorbs the momentum of liquid and gas and thus aids the separation. This is the initial phase of separation of liquid and gas.

2) Gravity settling section

This is the major vessel body part of the separator. In this section, liquid droplets of size more than 100-140 μm fall to the gas-liquid interface while smaller liquid droplets remain suspended in the gas. The remaining liquid in the gas is then separated at mist extractor. Sizing of vessel depends upon the size of droplet that needs to be handled. If the vessel design is flawed and larger sized droplets reach the mist extractor, they can overload it.

3) Mist extractor section

In final section, the gas containing small droplets of liquid is passed through a coalescing element or mist extractor. The coalescing element provides a large amount of surface area to separate smaller droplets from gas. The direction of the gas changes a lot when it passes through the mist extractor, because of the greater mass the droplets can't keep up with the gas. The small liquid droplets collide with the mist extractor and the fall to the liquid collection section by gravity.

4) Liquid collection section

It is the section of the vessel in which a specific amount of liquid is retained and the quantity is governed by the retention time specified.

A.1.3. Equipment Description

There are various types of separators such as horizontal, vertical, spherical etc. Separators are designed and manufactured taking into account specific advantages and limitations of different configurations. The basic criterion for selection is optimized lifecycle cost for required design intent.

1) Horizontal Separators

The mixed flow of gas and liquid enters the separator and impacts on the inlet diverter causing a sudden change in momentum causing the preliminary separation. Then the larger droplets of liquids separate due to gravity over the gravity settling section and fall to the bottom liquid collection section. The liquid is provided enough retention time for the dissolved gases to escape from the liquid gathered in the bottom liquid collection section and to collect in the vapour space above. Smaller liquid droplets of diameter less than 100 μm are difficult to be separated in the gravity settling section thus the gas before exiting the vessel is passed through the mist extractor to remove them. The large surface area provided by the coalescing section of mist extractor enables gravity separation of smaller droplets.

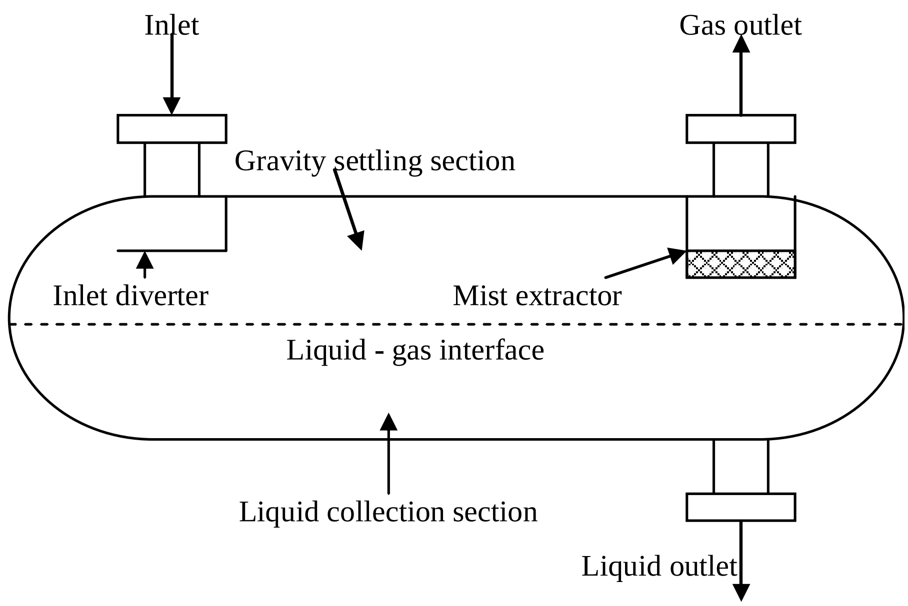


Figure A.1: Horizontal Separator Layout

The pressure in the separator is maintained by a pressure controller mounted on the gas outlet. The major advantage of horizontal separators over vertical is that they are compact and less expensive for required gas and liquid flow rate and are more effective in case of high gas-liquid ratio and foaming crude.

2) Vertical Separators

The mixed liquid and gas flow enters the vessel from side and impacts the inlet diverter. The initial gross separation happens at the inlet diverter. The liquid droplets fall down by gravity in the collection section. There are very few internal components in the collection section such as the vortex breaker for the outlet other than that it just a storage space for liquid designed as per the required retention time. Design should provide enough retention time so that liquid should reach equilibrium and dissolved gases should rise upward to the vapour space.

While rising up with the gases, the heavier liquid droplets fall to the collection section by gravity only droplets having very small diameter (approximately less than 100 μm) are usually carried by the gases till the mist extractor. When gas passes through the mist extractor these small droplets collide with the mist extractor coalescing section and fall downwards to the collection section by losing momentum.

Vertical separators are used for a flow with low or medium gas-liquid ration. A vertical separator fitted with a false cone at the bottom can be effectively used to separate heavier solid particles such as sand and other sediments.

A.1.4. Selection Considerations

Each type of separator has its own merits and demerits. Because of large liquid-gas interface in horizontal separator, it gives more opportunity for gases to escape as well provides more time for the liquid droplets to settle down. For given dimensions, horizontal separators have greater liquid capacity and better in case of liquid–liquid separation and foaming crude. However, horizontal separators can’t handle solid sediments as well as vertical separators. Vertical separators offer more liquid surge capacity than similar horizontal vessels for steady-state flow rate. Under normal conditions, horizontal separators are better for oil-gas separation for high gas-oil ratios. They are also better for handling problems with emulsions or foam. In case of low oil-gas ratio, vertical vessels are more effective. They can also be used as gas scrubbers where only fluid mists are required to be removed from the gas which demands extra surge capacity.

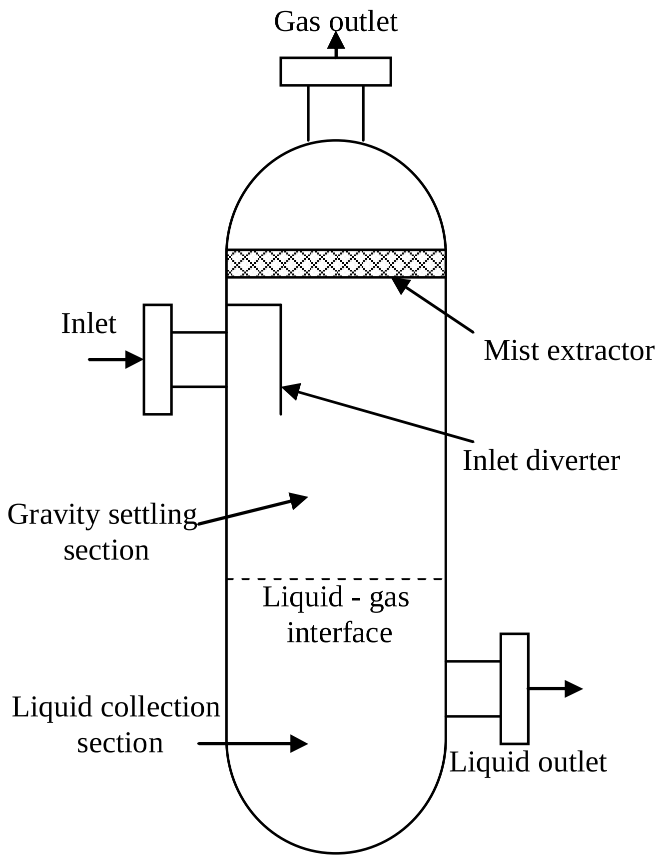


Figure A.2: Vertical Separator Layout

A.1.5. Vessel Internals

1) Inlet diverter

Inlet diverter as the name suggest is used to rapidly change the flow direction of the flow entering the separator for primary separation of oil and gas. It is essentially a baffle plate which can be a spherical dish, flat plate, angle iron, cone, elbow, etc in order for a rapid change in direction and velocity of the fluids and thus disengage the gas and liquid. Liquid particles in the same flow posses more energy than the gas as it has more density thus is can't change direction as rapidly as the gas. Thus by losing the momentum the heavier liquid particles get separated and fall to the bottom where as the gas tends to flow around the diverter.

2) Vortex breaker

While leaving the separator, the liquid has the tendency to form vortex or whirlpool by swirling at the nozzle entrance thus a vortex breaker is provided to prevent it when the liquid control valve is open. A vortex could be cause of forming foam or suck some gas out of the vapour space and re-entrain it in the liquid outlet. Vortex breaker are flat plates attached to the outlet nozzle opening parallel to the flow direction which prevents any circular motion and therefore tendency to form vortices is removed.

3) Mist extractor

Mist extractors are essentially coalescing element to separate smaller suspended liquid and solid particles from the gas. There are several types of mist extractors and the selection depends on

- Size of droplets the separator must remove.
- Pressure drop that can be tolerated in achieving the required level of removal.
- Susceptibility of the separator to plugging by solids, if solids are present.
- Liquid handling capability of the separator.
- Whether the mist extractor/eliminator can be installed inside existing equipment, or if it requires a standalone vessel instead
- Availability of the materials of construction that is comparable with the process.
- Cost of the mist extractor/eliminator itself and required vessels, piping, instrumentation, and utilities.

The most common type of mist extractor used in most of the separators is the knitted-wire-mesh type. They offer high surface area for very low volume. Instead of being woven rather than knitted, this type offers sufficient flexibility and structural stability. The wire-mesh mist extractors are often specified by certain thickness (usually 3–7 in.) and mesh density (usually 10–12 lb/ft3). They are usually made up of wires with diameters from 0.10 to 0.28 mm, with a typical void volume fraction of 0.95– 0.99. The mist extractor wire mesh pads are typically mounted at the separator gas outlet and are placed between support grids. The support grids have to have enough distance between then to allow sufficient flow area. Then the structure is mounted on support rings or frames to fasten it to the separator vessel. This type of mist extractors are usually installed in vertical upward gas flow. The effective working of the wire mesh type extractor largely depends upon the velocity of the gas passing through. At a very high velocity there is a possibility of liquid re-entering the gas and if the velocity is too low, the vapour just drifts through the mesh element without the droplets impinging and coalescing.

The selection for type of mist extractor involves a typical cost-benefit analysis. Wire mesh pads are the cheapest but are more prone to plugging. They have more deteriorating rate and mesh particles may get enter into the gas stream thus requires careful inspection and more frequent replacement. Vane type coalescing elements are more expensive but they are more robust and can work with higher velocity flows than the wire mesh. They are also less susceptible to plugging and deterioration than mesh pads. Microfiber units are the most expensive but are capable of capturing very small droplets. But they are most susceptible to susceptible to plugging. Fluid characteristics such as the droplet size to be removed largely affect the sizing and type of the mist extractor.

A.1.6. Potential Operating Problems

1) Foamy Crude

The crude from the well contains a lot of other impurities than gas and water such as CO_2 sand and other sediments, which are impractical to remove before the stream reaches the separator. These impurities mixed with water and oil causing foam. One impurity that almost always causes foam is CO2. Provided adequate retention time and liquid capacity foam does not provide any problem in separator as it gets dispersed on the mist extractor, but in a poorly designed vessel it can clog the mist extractor because of inadequate coalescing surface.

The major problems caused by the presence of foam are:

- As the liquid level control has to deal with foam, the control device operation is compromised since it has to deal with three liquid phases instead of two.
- Foam occupies a lot of vessel space as it has a high volume to weight thus reducing the liquid and gas carrying capacity of the separator.
- In case of excessive foam it becomes impossible to separate it from oil and gas before it exits the separator.

To get rid of foam, foam depressant chemicals are added to well fluid before entering the separator. Also to deal with foaming separator is designed for more capacity than the required. Foam depressants are not a reliable solution since well fluid composition changes with time and also the cost of foam depressants for high-rate production can be an issue.

2) Paraffin

Paraffin accumulation greatly reduces the efficiency of a separator. Vessel internals such as the coalescing elements in of the mist extractors are prone to plugging by accumulations of paraffin. For well fluid which poses the potential problem of paraffin, rather than a mesh type coalescing element, a plate type or a centrifugal mist extractor should be used. Separator internals such as man-ways, hand holes, and nozzles should be provided to allow steam, solvent, or other types of cleaning.

3) Sand

Sand is a potential hindrance for the proper working of separators at full potential. It causes blocking valves, vessel internals and accumulates at the bottom of the separator occupying the liquid collection section and thus reducing the retention time. Sand can be removed time to time by injecting water or steam in order to suspend it and drain from the bottom. In order to deal with the sand effectively and to drain it, a vertical separator can be fitted with a cone bottom. Cone is used only if sand is a prominent problem. The cone angle is usually between 45 to 60 degrees to the horizontal.

4) Liquid Carryover

During separation process if there is large amount of suspended liquid left in the gas while it leaves the separator, it is called as liquid carryover. Major reason for liquid carryover is poorly designed separator vessel or operating the separator over the specified limit of input flow rate. Thus a properly designed vessel with enough liquid and gas capacity and retention time and also by equipping the separator with a level safety high sensor to shut the inlet in case of excess flow can solve the problem of liquid carryover.

5) Gas Blowby

When excess gas escapes with the liquid through the liquid outlet, it is called as gas blowby. There can be number of reasons of gas blowby low liquid level, vortexing, or level control failure. Gas blowby could lead to a very dangerous situation. In case of gas blowby, the gas from the separator escapes and reaches the downstream equipments such as pumps which if not equipped for gas blowby can be damaged thus leading to a potential failure of entire facility. Normally a low level safety sensor for the liquid in the separator is installed in order close the liquid outlet or flow inlet. These sensors are set to close the inlet or outlet when the liquid level drops by about 10 to 15 % below the operating level.

6) Liquid Slugs

Liquid accumulates in the low spots of the pipelines and sometimes raises high enough to block the gas flow. This block then gets pushed by the gas as slug along the line. The volume or size of the liquid slug depends on the flow rates, flow properties, length and diameter of the flow line, and the elevation change involved. It is essential to spot the liquid slug locations before installing the pipelines. Also to prevent slug it is a normal practice to install a higher diameter pipe than the required size.

A.2. Basic Layout

Figure A.3 and Figure A.4 shows the basic dimensional layout of vertical and horizontal separators.

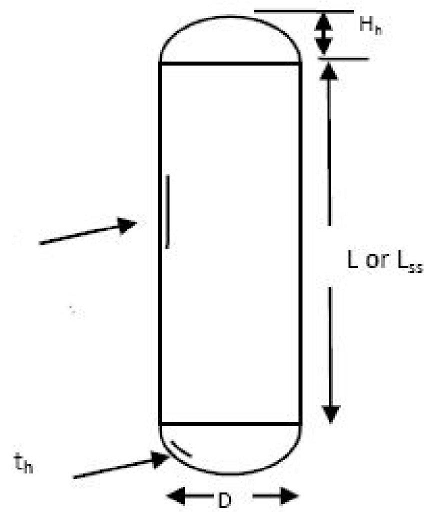


Figure A.3: Vertical separator Dimensions

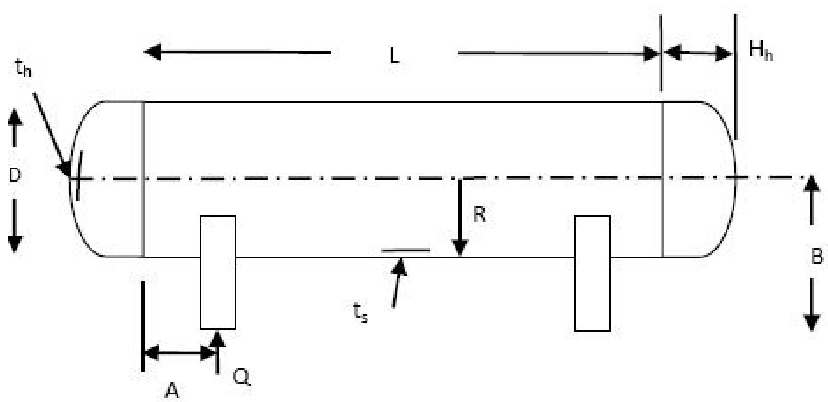


Figure A.4: Horizontal separator Dimensions

A.3. Design Process

A.3.1. Design of Separator Vessel

The separator is designed according to American Society of Mechanical Engineers' Boiler and Pressure vessel Code (ASME Code) section VIII. Pressure vessels are designed to withstand the loadings exerted by internal and external pressure, weight of the vessel, reaction of support, and impact. Temperature, pressure, feed composition and its mass flow rate is considered to select type and design of vessel and to come up with the dimensions of vessel. For this particular study the only load considered are internal pressure and temperature. Vessel size is decided depending upon the flow rate requirements.

A.3.2. Design Pressure

The design pressure for the vessel is called "Maximum Allowable working Pressure" (MAWP). The rules for setting the MAWP are given in table 6.1.

Table A.1: Maximum Allowable Working Pressure (MAWP)

Maximum operating pressure (psi)	MAWP (psi)
Less than 64.7	P + 10
64.8-264.7	P + 25
264.8-514.7	1.1 × P
514.8-1014.7	P + 50
1014.8 and higher	1.05 × P

A.3.3. Design of a Separator Flow Details

First calculate the density of liquid (oil), which can be calculated by the following formula.

$$\rho l = \frac{141.5}{131.5 + API} \tag{A.1}$$

Molecular weight (*MW*) of the gas can be found as the specific gravity (*SGg*) of gas is given by the following equation.

$$MW = 29 \times SGg \tag{A.2}$$

The compressibility factor is assumed as,

$$Z = 0.84 \tag{A.3}$$

Compressibility factor (Z) value is required to calculate the density of gas (ρg) by the following formula.

$$\rho g = \frac{2.7 \times SGg \times P}{T \times Z} \tag{A.4}$$

The drop diameter size (dm) is taken as input from user.

Drag coefficient (CD), which has been found to be a function of the shape of the particle and Reynolds Number of the flowing gas. For the purpose particle shape is considered to be solid, rigid sphere.

$$CD = \left[\left(\frac{24}{RE} \right) + \left(\frac{3}{RE} \right)^{0.5} + 0.34 \right] \tag{A.5}$$

Where, Reynolds Number (RE) is given by the following formula.

$$RE = 0.0049 \left(\frac{\rho g \times dm \times vt}{\mu} \right) \tag{A.6}$$

In this form, a trial-and-error solution is required since both particle size dm and terminal velocity vt are involved. Where, terminal velocity is given by the following equation.

$$vt = 0.0119 \left[\left(\frac{\rho l - \rho g}{\rho g} \right) \frac{dm}{CD} \right]^{0.5} \tag{A.7}$$

To get constant value for drag coefficient, value is assumed to find terminal velocity and Reynolds Number and which will be used in getting drag coefficient. This cycle will take place until all parameters become constant.

A.3.4. Design for Separator Shell Dimensions

Vertical Separator

The shell diameter is calculated as follows for gas capacity considering the required height of liquid, any diameter greater than the minimum required for gas capacity can be chosen.

$$D = \sqrt{5040\left(\frac{T \times Z \times Qg}{P}\right)\left[\left(\frac{\rho g}{\rho l - \rho g}\right)\frac{CD}{dm}\right]^{0.5}} \quad \text{(A.8)}$$

For liquid capacity to remain in the vessel the term used is called retention time tr in minutes. The liquid retention time requirement specifies a combination of diameter and liquid height and is taken as input from user. The height of the liquid (h) in inches is calculated by the following formula.

$$h = \frac{tr \times Ql}{0.12 \times D^2} \quad \text{(A.9)}$$

Calculate seam to seam length (*Lss*) in ft. The seam to seam length *Lss* of the vessel should be determined from the geometry of the vessel once a diameter is known.

$$Lss = h + 76 \quad \text{(A.10)}$$

Once shell diameter and seam to seam length is calculated, next standard sizes are selected. Selection of diameter and length depends upon slenderness ratio which is the ratio of shell length to shell diameter. For stability and structural rigidity, slenderness ratio is maintained in the range of 3 to 4.

If the ratio value is greater than 4, then next standard values are assumed until the required ratio is obtained,

Head depth *Hh* in inches for spherical head (considered for this study) is given by the following equation.

$$Hh = \frac{D}{2} \quad \text{(A.11)}$$

Where, D is the diameter of the vessel in inches.

Horizontal Separator

Effective length of vessel is calculated by the following formula to satisfy the gas capacity constraint.

$$Leff = \frac{420 \times T \times Z \times Qg}{D \times P}\left[\left(\frac{\rho g}{\rho l - \rho g}\right)\frac{CD}{dm}\right]^{0.5} \tag{A.12}$$

For design purpose the vessel is considered half filled at any given time.

To satisfy the Liquid capacity constraint, sufficient retention time should be provided to allow the liquid reach equilibrium.

$$Leff = \frac{tr \times Ql}{0.7D^2} \tag{A.13}$$

$$Lss = Leff + D \tag{A.14}$$

Once shell diameter and seam to seam length is calculated, next standard sizes are selected to compute slenderness ratio, which is required to be in the range of 3 to 4.

If the ratio value is greater than 4, then next standard values are assumed until the required ratio is obtained.

Head depth *Hh* in inches for spherical head (considered for this study) is given by the following equation.

$$Hh = \frac{D}{2} \tag{A.11}$$

Where, D is the diameter of the vessel in inches.

A.3.5. Design of Separator Details

Required material properties obtained for user are Yield strength (*Sy*) and Poisson's ratio. The vessels are designed in accordance with Division 1 rules thus the factor of safety considered is 4. Thus allowable stress value for shell is,

$$Ss = Sh = Sn = \frac{Sy}{4} \quad (A.15)$$

The required joint efficiency (*E*) is obtained from the user.

The required corrosion allowance (*CA*) is also taken as a user input.

Shell thickness *ts* (in) can be found as per ASME division VIII and given by,

$$ts = \frac{P \times D}{2[(Ss \times E) - (0.6 \times P)]} + CA \quad (A.16)$$

Head thickness *th* in inches as per ASME code for vessel is calculated by the following formula. Where *Sh* is a maximum allowable stress in psi for the material used in head which in this study considered same as the shell material (*S*).

$$th = \frac{P \times D}{(2 \times Sh \times E) - (0.2 \times P)} + CA \quad (A.17)$$

Shell thickness and head thickness is then selected as the next higher value from the standard commercially available plate sizes in inch given below.

The inlet and outlet nozzles are selected according to the flow rate required as below.

$$dn = \sqrt{\frac{4 \times Qf}{\pi \times vin}} \quad (A.18)$$

The flow rate is calculated as,

$$Qf = Ql + (166.67 \times Qg) \quad (A.19)$$

Nozzle thickness is also selected in accordance with ASME vessel design code as below. The nozzle is assumed to be made of same material as the shell.

$$tn = \frac{P \times D}{2[(Sn \times E) - (0.6 \times P)]} + CA \qquad (A.20)$$

Once thickness obtained, the required nozzle is selected from the standard commercially available pipe sizes below with required standard thickness mentioned above.

Next the weight of the vessel is calculated. Vessel weight is required for vessel support design as well as to determine the foundation requirements. The shell weight is calculated as,

$$Ws = 11 \times D \times ts \times Lss \qquad (A.21)$$

The head weight is calculated as,

$$Wh = 2[(0.34 \times th \times D^2) + (th \times D)] \qquad (A.22)$$

The total weight of the vessel is given as,

$$Wv = Ws + (2 \times Wh) \qquad (A.23)$$

The weight of the internals is assumed to be 10 % of the total vessel weight.

$$W\,\text{int} = 0.1 \times Wv \qquad (A.24)$$

Operating weight *Wo* in lbs is the erection weight of the tower with full water and it is used in calculating the thickness of the vessel support.

$$W = Wv + W\,\text{int} + Wo \qquad (A.25)$$

A.3.6. Design of Separator Support

Vertical Separator

Height of the skirt *hT* is calculated in inch by the formula,

$$hT = (0.25 \times D) + 24 \quad \text{(A.26)}$$

The required thickness of the skirt is calculated as,

$$tskirt = \frac{W}{(D + 2ts) \times 3.124 \times Sh \times E} \quad \text{(A.27)}$$

The vertical vessels are fastened to the concrete foundation, skid or other structural frame by mean of anchor bolts and the base (bearing) ring. An approximate method based on preapproved design results of will be used for anchor bolts and base design.

The bolt circle diameter is calculated as below.

$$d = D \times 1.15 \quad \text{(A.28)}$$

The inside and the outer diameter for the base plate is given by,

$$Di = D - 2 \quad \text{(A.29)}$$

$$Do = 2d - Di \quad \text{(A.30)}$$

The base plate thickness is calculated using following equation and once value obtained, next standard value of standard available sheet thickness is selected.

$$tb = 0.321 \frac{Do - d - ts}{2} \quad \text{(A.31)}$$

For given bolt circle diameter d in inches, find the area within the bolt circle *BA* in sq-in and the circumference of bolt circle *CB* in inches.

$$CB = 2 \times \pi \times r \quad \text{(A.32)}$$

It is preferred to use minimum eight anchor bolts to fasten the vessel to the foundation.

The bolt material selected is the standard SA325, which is *Sb* = 15000 psi.

Determine the maximum tension Tb on bolt circle in lb/in^2.

$$Tb = -\frac{W}{CB} \tag{A.33}$$

Calculate the required area of one bolt *BA* (sq-in).

$$BA = \frac{TB \times CB}{Sb \times N} \tag{A.34}$$

Once bolt area is obtained, the required bolt diameter is calculated. The next bolt size from the standard bolt sizes below is selected.

Horizontal Separator

1) The contact angle for the saddle is assumed to be 180^{O}
2) For given study since the vessel diameter is below 90 inch, the saddle does not have any web flanges.
3) The vessel is considered to have two saddle supports at a length of 0.25×lss from either faces of shell as shown below by dimension "A" shown in the basic layout.
4) Dimensions of the support saddle based the diameter of the vessel and are given in table below. All values are in inch.

A.3.7. Separator Internal Details

Inlet diverter dimensions depend upon the dimensions of the inlet nozzle diameter. The dimensions are given by,

Inlet diverter length,

$$idl = 4 \times dn \tag{A.35}$$

Inlet diverter extension from the wall of the vessel,

$$ide = dn \quad (\text{A.36})$$

Inlet diverter thickness,

$$idt = 0.5 \text{ in}$$

Vortex breaker dimensions are calculated using the outlet nozzle diameter as below.

Vortex breaker length,

$$vdl = 4 \times dn \quad (\text{A.36})$$

Vortex breaker extension from vessel wall,

$$vde = dn \quad (\text{A.37})$$

Vortex breaker thickness,

$$vdt = 0.5 \text{ in}$$

Mist extractor dimensions are usually of standard sizes and the mist extractor support dimension is calculated using shell dimensions.

Mist extractor mesh element thickness,

$$MEt = 6 \text{ in}$$

Mist extractor support thickness

$$MESt = ts \quad (\text{A.38})$$

Mist extractor support extension length from the vessel wall.

$$MESe = D / 8 \quad (\text{A.39})$$

Appendix B

Standard Component Sizes

Table B.1: Standard sizes and combinations for shell diameter (*D*) and shell length (*Lss*) in inches

D × *Lss*	*D* × *Lss*	*D* × *Lss*	*D* × *Lss*	*D* × *Lss*
16 × 60	20 × 60	24 × 60	30 × 60	36 × 60
16 × 90	20 × 90	24 × 90	30 × 90	36 × 90
16 × 120	20 × 120	24 × 120	30 × 120	36 × 120
42 × 90	48 × 90	54 × 90	60 × 180	36 × 180
42 × 120	48 × 120	54 × 120		
42 × 180	48 × 180	54 × 180		

Table B.2: Standard metal sheet thickness in inches

0.188	0.438	1.125	2.0	3.625
0.219	0.469	1.188	2.25	3.75
0.257	0.5	1.25	2.375	3.875
0.281	0.563	1.313	1.188	4.0
0.313	0.750	1.375	2.5	4.125
0.344	0.813	1.438	2.625	2.25
0.375	0.875	1.55	3.375	
0.406	1.0	1.625	3.5	
0.625	1.755			
0.688	1.875			

Table B.3: Standard nozzle (pipe) sizes in inches

2
3
4
6
8
10
12
14
16
18
20

Table B.4: Standard bolt size for vessel supports in inches

0.5
0.625
0.75
1
1.125
1.25
1.375
1.5
1.625
1.75
1.875
2
2.25
2.5

Table B.5: Horizontal separator saddle dimensions

Shell diameter (D)	Saddle base length (A)	Height of vessel centreline from ground (B)	Saddle base width (C)	Saddle width at vessel end (D)
16	14	13	4	4
20	17.5	16	4	4
24	21	19	4	6
30	28	24	4	6
36	30.5	29	6	11
42	36.5	34	6	11
48	42	38	6	11
54	47	43	6	11
60	52	48	6	11

Support bolt location (E)	Saddle base thickness (G)	Side-plate and rib thickness (H)	Saddle ring thickness (K)	Support bolt diameter
5	0.25	1	0.25	0.5
6.5	0.25	1	0.25	0.5
7.5	0.25	1	0.25	0.5
9	0.5	1	0.25	0.5
11	0.5	1	0.25	0.5
14	0.5	1.5	0.25	0.75
16	0.75	1.5	0.25	0.75
18	0.75	1.5	0.25	0.75
20	0.75	1.5	0.25	0.75

Appendix C

Pseudo Codes

```
 Data:
  Operational parameters;
  Geometric parameters;
  Mesh physical properties;
  Material properties;
  Functional parameters;
  Constraints parameters;
  Standard sizes of components (from external data structure);
```

```
 /*Algorithm*/
Main {
 Generation of parameters using engineering design formulae()
 {
      Calculation_of_shell_body parameters()
       {
       //follow engineering formuliea as discussed in Appendix A
       //…
       }
      Calc_vessel_head_parameters();
       //...
       Expot_parameters("D:\\Final\\Vertical\\CDMV.exp”);
 }
 Node mid-plane_modeling:
 Generation of mid-plane CAD model parametrically through API ();
 Generation of mid-plane finite element mesh parametrically through
 API();
 Generation of mid-plane CAE analysis parametrically through API();
 Provide the analysis results to user();
 if (analysis stress and deformation values are not within allowable
 limits)
      {
      recalculate parameters according to new requirements();
      maintain the history by creating versioned record of CDM files();
      goto Node mid-plane_modeling;
 }
 Node detailed_solid_modeling:
 Generation detailed solid CAD model parametrically through API();
 Generation detailed solid finite element mesh parametrically with API();
 Generation detailed solid of CAE analysis parametrically through API();
 Provide the analysis results to user();
 if (analysis stress and deformation values are not within allowable
 limits)
      {
      recalculate parameters according to new requirements();
      maintain the history by creating versioned record of CDM files();
             goto Node detailed_solid_modeling;
      }
 submit the final versioned CDM, CAD, FE and CAE models to user();
 }
```

Figure C.1: Pseudo-code structure for overall design process

```
Data:
 Geometric parameters;
 Constraints parameters;
 Standard sizes of components (from external data base);
```

```
/*Algorithm*/
Main {
 /* Start new NX session*/
 Session *theSession = Session::GetSession();
 FileNew1->SetNewFileName("D:\\Final\\Vertical\\cad3d.prt");
 ...

 /* Import parametric data in the form of expressions */
 markId4 = theSession->SetUndoMark(Session::MarkVisibilityVisible,
 "Expression");
 bool expModified1;
 std::vector<NXString> errorMessages1; workPart->Expressions()->
 ImportFromFile
 ("D:\\Final\\Vertical\\CDMV.exp",ExpressionCollection::ImportModeRepla
 ce;
 ...

 /* Create vessel body structure */
 cylinderBuilder1->Diameter()->SetRightHandSide
 ("Shell_diameter+(2*Shell_thickness)");
 cylinderBuilder1->Height()->SetRightHandSide("Shell_length");
 theSession->SetUndoMarkName(markId5, "Cylinder Dialog");
 ...

 /*Create head structure, nozzle structures and internal details*/
 ...

 /* Save the model*/
 PartSaveStatus *partSaveStatus1;
 partSaveStatus1 = workPart->Save(BasePart::SaveComponentsTrue,
 BasePart::CloseAfterSaveFalse);
 delete partSaveStatus1;

 /* Terminate NX session*/
 UF_CALL(UF_terminate());

 }
```

Figure C.2: Pseudo-code structure for CAD model generation

```
Data:
 Material parameters;
 Functional details;
 Preconstructed CAD model as geometric input;
```

```
/*Algorithm*/
Main {
 /* Start new NX session*/
 Session *theSession = Session::GetSession();
 fileNew1->SetNewFileName("D:\\Final\\Vertical\\fem3d.fem");

 /* Import CAD model for geometry input */
 PartLoadStatus *partLoadStatus1;
 basePart3 = theSession->Parts()-
 >OpenBaseDisplay("D:\\Final\\Vertical\\cad3d.prt",
 &partLoadStatus1);
 ...

 /* Import parametric data in the form of expressions */
 markId14 = theSession->SetUndoMark(Session::MarkVisibilityVisible,
 "Expression");
 bool expModified1;
 std::vector<NXString> errorMessages1;
 workFemPart->Expressions()-
 >ImportFromFile("D:\\Final\\Vertical\\CDMV.exp",
 ExpressionCollection::ImportModeReplace, &expModified1,
 errorMessages1);
 ...

 /* Create a finite element mesh*/
 mesh3dTetBuilder1 = meshManager1-
 >CreateMesh3dTetBuilder(nullCAE_Mesh3d);
 theSession->SetUndoMarkName(markId6, "3D Tetrahedral Mesh Dialog");
 ...

 /* Assign mesh physical properties */
 CAE::PropertyTable *propertyTable1;
 propertyTable1 = mesh3dTetBuilder1->PropertyTable();
 propertyTable2->SetScalarPropertyValue("quad mesh overall edge size",
 expression1);
 ...

 /* Save the model*/
 PartSaveStatus *partSaveStatus1;
 partSaveStatus1 = workPart->Save(BasePart::SaveComponentsTrue,
 BasePart::CloseAfterSaveFalse);
 delete partSaveStatus1;

 /* Terminate NX session*/
 UF_CALL(UF_terminate());

 }
```

Figure C.3: Pseudo-code structure for FE mesh model generation

```
/*Algorithm*/
Main {
 /* Start new NX session*/
 Session *theSession = Session::GetSession();
 fileNew1->SetNewFileName("D:\\Final\\Vertical\\sim3d.sim");

 /* Import FE model for mesh input */
 PartLoadStatus *partLoadStatus1;
 basePart3 = theSession->Parts()-
 >OpenBaseDisplay("D:\\Final\\Vertical\\fem3d.fem",
 &partLoadStatus1);
 ...

 /* Import parametric data in the form of expressions */
 markId14 = theSession->SetUndoMark(Session::MarkVisibilityVisible,
 "Expression");
 bool expModified1;
 std::vector<NXString> errorMessages1;
 workSimPart->Expressions()-
 >ImportFromFile("D:\\Final\\Vertical\\CDMV.exp",
 ExpressionCollection::ImportModeReplace, &expModified1,
 errorMessages1);
 ...

 /* Apply constraints to the model*/
 markId15 = theSession->SetUndoMark(Session::MarkVisibilityVisible,
 "Start");
 theSession->SetUndoMarkName(markId15, "Fixed Constraint Dialog");
 CAE::SimBCBuilder *simBCBuilder2;
 simBCBuilder2 = simSimulation3->CreateBcBuilderForConstraintDescriptor
 ("fixedConstraint", "Fixed(2)");
 ...

 /* Assign functional and operational properties */
 autoBCBuilder1 = simSimulation4->CreateAutoBcBuilder
 ("Surface to Surface Gluing", "Face Gluing");
 ...
 CAE::SimBCBuilder *simBCBuilder3;
 simBCBuilder3 = simSimulation4->CreateBcBuilderForLoadDescriptor
 ("2D3DFaceNormalPressure", "Pressure(1)");
 theSession->SetUndoMarkName(markId19, "Pressure Dialog");
 ...

 /*Solve the model for solution*/
 simSolution1->Solve(CAE::SimSolution::SolveOptionSolve,
 CAE::SimSolution::SetupCheckOptionCompleteCheckAndOutputErrors);
 ...

 /* Save the model*/
 PartSaveStatus *partSaveStatus1;
 partSaveStatus1 = workPart->Save(BasePart::SaveComponentsTrue,
 BasePart::CloseAfterSaveFalse);
 delete partSaveStatus1;

 /* Terminate NX session*/
 UF_CALL(UF_terminate());
}
```

Figure C.4: Pseudo-code structure for CAE analysis