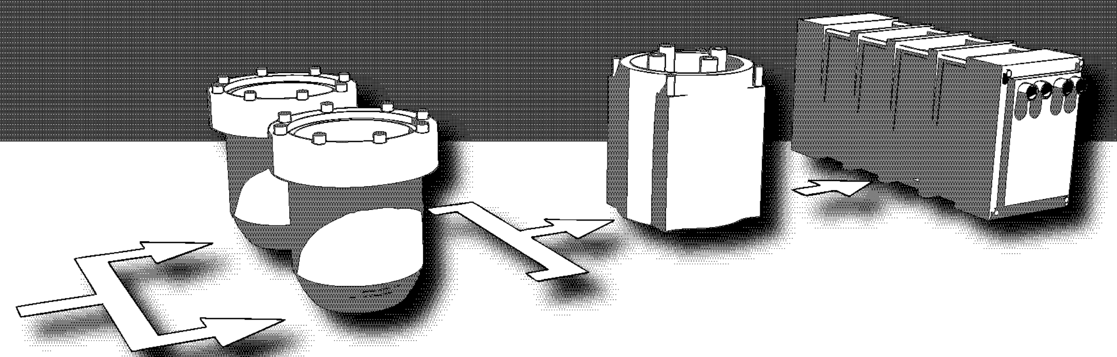
Diss. ETH No. 16694
ISBN 3-906734-46-3
www.mosi.sk

Deterministic & stochastic batch design optimization techniques

Andrej Mošať

16694

DETERMINISTIC AND STOCHASTIC BATCH DESIGN OPTIMIZATION TECHNIQUES



Andrej Mošať

Diss. ETH No. 16694

Diss. ETH No. 16694
ISBN 3-906734-46-3

Deterministic and stochastic batch design optimization techniques

A dissertation submitted to the
SWISS FEDERAL INSTITUTE OF TECHNOLOGY ZURICH

for the degree of
Doctor of Sciences

presented by
Andrej Mošať
Ing., Slovak Technical University Bratislava
born December 14th, 1978
citizen of Slovak Republic

Accepted on the recommendation of
Prof. Konrad Hungerbühler, examiner
Prof. Eckart Zitzler, co-examiner
Dr. Ulrich Fischer, co-examiner

Zurich 2006

Janke

Acknowledgments

First off, I would like to thank Prof. Konrad Hungerbühler for his helpful insights and discussions. Thank you, Konrad for beign a stable lighthouse in the fog.

I would like to thank Prof. Eckart Zitzler for great lessons on evolutionary algorithms and for precising this book.

Thanks to Dr. Ulrich Fischer and his dedication, the countless razor-sharp advices turned into fruitful texts, posters, presentations and methods. The effort you put into the research, education and the way of managing personal relationships with flying colours, often under heavy pressure, is tremendous. Thank you for being an editor, a confidant and a friend.

Mr. Dr. Laurent Cavin from Puy, nor R. Heinlein, I. Asimov or C. Clark could express with their master style the extraordinary amount of gratitude for the patience, humour, knowledge, advices and occassional Latin quotations, exactly in this order, you provided on your journey through Hoenggerberg. It was a once-in-a-lifetime pleasure to meet you and discuss any topic imaginable. Thanks to you, the words like 'diplomacy', 'humour-on-the-edge' or 'gourmand' are slowly diffusing into my dictionary and are not foreign anymore.

Especial thanks to the staff at Lonza, AG, Dr. Zimmermann, Dr. Lange, Dr. Zaupa, Dr. Capelli, Dr. Dubuis, Dr. Keuth for supporting me from the beginning in a very proffessional way and being very kind and patient.

I would also like to thank cordially my loving fiancée Janka for the incredible moral support and a wish-come-true care.

Ing. Andrej Szíjjárto, I thank you from my heart for all the wild trips, the sports, the political discussions, the driving, the learning, the sharing and having all kinds of marvelous fun together. True friends are difficult to find.

To mention here all the interesting and helpful persons and friends who assisted me in the last four years is an incredible task. Let's just say, that you know how great you are and that your support shall never be forgotten. Thank you very much.

Last, but not least, I'd like to thank my parents and Mr. God for everything.

Contents

List of Figures

List of Tables

Abstract

Custom chemicals and pharmaceuticals manufacturing is performed very often in multipurpose batch plants. Multipurpose plants offer a high degree of flexibility for the design and operation of chemical processes. As a consequence, problems related to mathematical models of different scenarios such as: design of a single process, design of a new plant, retrofit of an existing plant or scheduling a given portfolio of processes in a given plant are complex and demand substantial computational resources. The mathematical models of a given problem related to batch processing are usually very difficult to formulate. Typically these problems are highly nonlinear, and unless simplified, are believed not to be solvable in polynomial time. The problem complexity even increases if multiple objectives are to be considered. On top of the multiobjective decision problems, the uncertain factors inclusion further complicates the mathematical models.

A solution to the discussed problems is a design, i.e. an allocation of recipe tasks to batch plant equipment. The aim of the proposed algorithms is to optimize the design of a single chemical process to be implemented in an existing multipurpose batch plant. Various objective functions used in the multiobjective optimization are defined as quantitative measures of the quality of such process designs. In this work a Tabu Search metaheuristic method was successfully applied to a wide range of problems.

The main goals of the research posed in this thesis can be summarized as:

1. Formulation of the mathematical problems and development of methodologies supporting the batch plant engineering development team during the process design phase.
2. Proposing a set of solving algorithms for tackling the presented problems.
3. Finding and demonstrating a practical method of reducing significantly the size of the batch design optimization domain. A *Superequipment concept* was developed as a mathematical model of an equipment class capable of performing any chemical operation class.
   A *Superequipment unit* must additionally fulfill the reality criterion, that means it must be transformable into a real equipment unit during and after the optimization. This concept reduces the combinatorial complexity in the solution space significantly.
4. Stipulating and investigating a stochastic mathematical model, related to batch process development, which can be handled by multiobjective optimization in a reasonable time frame allowing rapid result output. Productivity robustness related to a design is defined in the stochastic approach.
5. Automatically selecting a set of feasible and good-performing designs as a basis for the decision making in the early batch process development.

The problem domain of early stage batch process development is extended in the presented formulations. In addition to the optimization of a single deterministic design, the new optimization algorithms assist in: retrofitting of equipment, grass-root design of a new plant, automated production plant line selection or in evaluating a design robustness by an automated stochastic method. The results compilation is also automatically presented as a selection of individual designs, sorted according to prioritized list of objective function values.

The methods, demonstrated on various case studies, show feasibility of the resulting designs, a broad applicability of the methods for automatized integrated process development. The discussed stochastic batch process design approach presents important measures related to the risks in the preliminary design stages.

Zusammenfassung

Die Prozessentwicklung ist ein grundlegender Teil eines industriellen Projekts. Pharmazeutika und Spezialitätenchemie werden meistens in Batch-Anlagen hergestellt. Die Notwendigkeit einer effizienten Prozessplanung stellt den Prozessdesigner vor die Aufgabe, neue automatisierte Methoden zu entwickeln.

Die Methoden sollen eine *Integrierte Prozessentwicklung*, das heißt Beurteilung von allen relevanten Aspekten des Prozesses gewährleisten. Dabei ist zu beachten, dass oft nur unvollständige Informationen über das Verfahren vorhanden sind. Die Applikation dieser Methoden in der Praxis sollte eine schnelle Übersicht von Prozessalternativen und deren Parametern liefern. Eine methodische Formulierung von mathematischen Modellen für die Batch-Prozesse ist problematisch. Zur Zeit gibt es keine rigorosen Algorithmen, die eine zuverlässige Lösung für Probleme dieser Art liefern können. Üblicherweise werden die Probleme vereinfacht oder mit Hilfe von *„random – search"* Algorithmen gelöst, die keine optimalen Lösungen garantieren. Die Komplexität der Probleme steigt, wenn man mehrere Zielgrößen betrachtet oder wenn Unsicherheiten in die Problemstellung mit einbezogen werden.

Das Hauptziel dieser Arbeit liegt in der Entwicklung von Methoden und Software für die frühe Beurteilung der Prozessalternativen in Mehrzweckanlagen. Die Methoden sollen vielversprechende Prozessdesigns entwerfen, deren Zielgrößen berechnen und eine Auswahl von Designs zur Weiterentwicklung vorschlagen.

Die Basis für die Entwicklung dieser Methoden bildet eine Reihe von Problemstellungen bezüglich der Prozessentwicklung in Batchanlagen.

In dieser Arbeit wurden die folgenden Themen behandelt: eine deterministische Problemstellung, eine stochastische Problemstellung mit *Design Robustness*, das *Superequipment Konzept.*

Eine deterministische Problemstellung baut auf mathematischen Modellen auf, welche die Daten über die Batchanlage, das Rezept und die Produktionsheuristiken enthalten. Damit kann man eine Reihe von Szenarien bezüglich des Designs eines Prozesses, das Retrofit von Produktionsanlagen oder Neuanlagengestaltung lösen.

Ein weiterer Lösungsansatz ist das *Superequipment Konzept.* Das Superequipment ist ein mathematisches Modell. Es ist definiert als eine übergeordnete Klasse von Produktionsgeräten. Diese Klasse kann alle betriebsüblichen Operationen durchführen. Dabei muss es aber die Bedingung der Realität erfüllen, so dass dieses Modell innerhalb der Optimierung in ein reales Gerät transformierbar ist. Letztendlich reduziert dieses Konzept die Komplexität von Optimierungsproblemen im Prozessdesign. In dieser Arbeit wird das Superequipment Konzept in unterschiedlichen Fallstudien erfolgreich demonstriert.

Die nächste Stufe in der Entwicklung der präsentierten Methoden ist die Betrachtung einer Batchproduktion mit multiplen Entscheidungskriterien unter Beachtung von Unsicherheiten. Das Ziel ist, mehrere Designs zu entwurfen, die nicht nur eine gute Produktionsrate haben, sondern auch die erwartete Produktionsrate unter variablen Beschaffenheiten gewährleisten können. Die Robustheit wird als ein Kriterium für das Design vorgeschlagen. Die unsicheren Parameter werden mit der Latin-Hypercube Methode behandelt und sind im Rezept definiert. In der Evaluationsfunktion werden dann die Wahrscheinlichkeitsdistributionen der Variablen für die Berechnung der Produktionsrate, und die Produktionsrate zur Berechnung der Robustheit benutzt.

Um die Methodologie ausführlich zu testen, wurden mehrere industrierelevante Fallbeispiele verwendet. Listen von vielversprechenden Designs mit entsprechenden Beschreibungsindikatoren wurden innerhalb einer Stunde Rechenzeit generiert. Die verschiedenen Indikatoren erlauben Diskussionen und Bewertungen des Designs bezüglich weiterer Kriterien (z.B. Batchgrösse), die während der Optimierung nicht benutzt wurden. Die Lösunganätze von unterschiedlichen kombinatorischen Problemstellungen kann man erfolgreich mit dem Superequipment Konzept verbessern. Die Fallbeispiele haben gezeigt, dass eine Rechenzeitersparniss gegenüber den konventionellen Methoden erreicht werden kann.

Die präsentierte Unsicherheitsbehandlung beider multikriterieller Optimierung demonstriert anhand mehrerer Fallbeispiele, welche der Prozessparameter einen bedeutenden Einfluss auf die Designproduktionsrate haben.

Die Methoden, die in dieser Arbeit vorgestellt werden, zeigen eine gute Anwendbarkeit, relativ niedrige Rechenzeiten und helfen bei den Entscheidungen im Rahmen der Integrierter Prozessentwicklung.

Introduction

1

This chapter states the problems encountered in batch process design, presents an overview of literature discussing such problems, reviews the process of selection of suitable optimization algorithm and in the final phase presents the research goals tackled in this thesis.

1. 1 Integrated batch process development

Speciality chemicals and pharmaceutical products are typically produced in batch processes. Custom batch manufacturing requires flexible response to varying amounts of product orders, delivery times and on demand production. The corresponding plants are often classified as multiproduct batch plants or as multipurpose batch plants.

According to (Rippin, 1983, Rippin, 1991) the terms *multipurpose batch plant* and *multiproduct batch plant* have been defined as follows:

multiproduct batch plant configuration implies using the plant in the mode where each product follows the same processing sequence through all the process steps.

multipurpose batch plant implies a configuration of the equipment, where each product follows its own distinct processing sequence. In the plant the connectivity among units is flexible. This flexibility then allows to use the production line for manufacturing more types of products with the possibility of simultaneous processing. Decisions are to be made as how to allocate the pieces of equipment and how to manage re-piping of the units.

In practice combinations of these two limiting scenarios might also arise. Multipurpose plants can be used in two main modes: either only one production runs in the plant at a given time or many processes run concurrently. Some multipurpose plants consist of discrete but flexible production lines that are independent from each other.

Because of the escalating importance of these types of chemical processes, in recent years increased research efforts have been undertaken to develop design methods for batch processes. Many methods deal with the grass root design of multiproduct or multipurpose batch plants and include the equipment sizing problem (Grossmann and Sargent, 1978, Papageorgaki and Reklaitis, 1990, Sparrow et al., 1975, Voudouris and Grossmann, 1996). In most cases, the authors consider the case where many productions run concurrently.

Relatively few publications have been presented that deal with the optimum design of a single batch process. For grass root design, (Loonkar and Robinson, 1970) described a procedure for the cost optimum design and apparatus sizing of a single batch process, while (Takamatsu et al., 1982) presented a similar approach that considers the possibility of intermediate storage. (Yeh and Reklaitis, 1987) presented a method for the preliminary grass root design of a single monoproduct batch process including an approximate sizing procedure. (Mauderli and Rippin, 1979) developed a method for planning and scheduling in multipurpose batch plants. While they consider many concurrent productions, their first step consists of the heuristic generation of design alternatives for the production of single products taking into consideration the plant specifications and the process requirements. (Wellons and Reklaitis, 1989) developed a MINLP formulation for such a design task. They generate groups of equipment units that can handle one particular step of the chemical process. The resulting combinatorial tree is optimized for the maximum production. The results are typically a list of batches (potentially of different batch size) that take different paths through the plant and run in a fixed sequence.

The pre-assignment of equipment units to a given process step is a constraint that limits the design possibilities in multipurpose plants in which equipment units can be used for several tasks and can thus be assigned to different process steps. (Wellons and Reklaitis, 1991) revised their design procedure for the scheduling of different processing paths for the same product and included the equipment assignment problem into the optimization. However, for control, safety and quality reasons as well as due to good manufacturing practice (GMP) regulations, subsequent batches are often preferred to follow the same path and to be of equal size. Under this perspective, the objective changes from the optimum schedule of n subsequent batches to the design of the single most efficient batch. To our best knowledge, the only such method presented so far is the one we described in (Cavin et al., 2004).

In order to identify an optimal solution for this problem it is important to consider in the design procedure all details and existing constraints such as equipment specifications (e.g. range of operating temperature and pressure, lining material, special supply pipes, the floor at which each equipment is located), design constraints (e.g. feasible and infeasible connection of equipment units), and process requirements (e.g. reaction mixture that cannot be safely transferred, thus forcing several operations to be conducted in the same equipment unit). This is the approach taken in the method presented here.

The most common approach in praxis and literature is a deterministic one (Balasubramanian and Grossmann, 2003a, Papageorgaki and Reklaitis, 1990, Mauderli and Rippin, 1979), in which fixed inputs are used for rendering fixed outputs for instance simulations where quality and delivery time requirements of each product must be fulfilled (see for instance (Biegler et al., 1997, Voudouris and Grossmann, 1996, Rippin, 1991)). However, it is often the case that many factors are unknown or uncertain in the early stage, for example yield of a reaction as a function of reaction time, problems with crystallization, which is often unpredictable in the scaled-up processing or solvent volume needed for a given task. These uncertainties call for a stochastic approach in optimizing the process design.

In the remainder of this section the literature on deterministic and stochastic batch design problems is reviewed.

1. 1.1 Deterministic batch design problem

The basic formulation and systematics of batch plant design problems was published by many authors, for example (Rippin, 1991, Rippin, 1983, Allgor et al., 1996, Allgor et al., 1999, Biegler et al., 1997, BarbosaPovoa and Macchietto, 1993, Reklaitis, 1989). In general the problem is to minimize costs and maximize productivity of the design layout for:
(a) a single process optimization focused on proposing a single optimal process layout, where each chemical task is assigned to a given equipment unit or, in a different formulation, a unit is assigned to a task
(b) scheduling, i. e. optimization of multiple processes running either in parallel

or successively in time, where the focus of optimization lies on fulfilling the market demand for a number of products under delivery time and other constraints.

In the aforementioned literature, the general aim of the multipurpose batch plant optimization is to provide an optimal schedule for subsequent batches in order to meet the required productivity, time and product amount constraints. The methods vary, but the basic assumption is that a given batch operation can be performed by different equipment in subsequent batches. Therefore the process follows multiple different path flows through the plant equipment. Due to safety and regulatory GMP reasons in pharmaceutic production, it is not allowed to modify the allocation of operation to equipment from batch to batch and the process flow path through the plant must remain constant for the whole campaign, or in the case of pharmaceuticals, for all campaigns after a FDA approval (FDA, 2006). Therefore the problem of scheduling of subsequent batches changes to finding an optimal design, which will remain unmodified during the whole campaign.

As (Rippin, 1991) refers, three essential components are presented in any batch processing problem:

- product requirements
- process tasks, which must be carried out in their appointed sequence to produce each of the products
- equipment needed for carrying out the processes

From these requirements, different problem settings might arise, for example:

- *grass-root design* of a new batch plant; papers describing this problem are taking major part of the publications, see for example (Espuna et al., 1989, Yi and Reklaitis, 2002, Wang et al., 1999, Yeh and Reklaitis, 1987).
  (Allgor et al., 1996, Allgor et al., 1999) define and solve a preliminary design of a batch plant and demonstrate the method on several case studies, considering thermodynamics, recipe related constraints, reaction specific data, equipment limitations and more.
- *retrofitting the batch plant* is today even more used than grass-root design (see (Papageorgaki and Reklaitis, 1993, Young and Reklaitis, 1989).
  (Barbosa-Povoa and Macchietto, 1994, Voudouris and Grossmann, 1996) have published an introduction to the problem structure, proposing tools and algorithms for solving the retrofit problem, where mostly MINLP methods for solving discrete optimization, and its variations are applied in:
  - handling specific product with well-known and established routes in the factory
  - designing the factory for the future requirements and products which may come to the production in the near future horizon
- *time-related problems*: scheduling of the process, see (Mockus and Reklaitis, 1999, Reklaitis, 1995, Wellons and Reklaitis, 1989). The problems include also the use of equipment under uncertainty with different approaches, see for example (Balasubramanian and Grossmann, 2003b, Balasubramanian and Grossmann, 2003a)
- *sizing the equipment and capacity related problems* (Terwiesch et al., 1994, Young and Reklaitis, 1989)
- *mode of operation of the plant*: equipment pool model, production lines model,

or other existing models (Rippin, 1993)

- *considering environmental impacts* of the manufacture, waste management algorithms (Hungerbühler et al., 1999, Stefanis et al., 1997, Dedieu et al., 2003)

Some of the tasks which are closely related to industrial problems of batch processing can be solved by help of optimization techniques. Usage of such methods allows for faster screening of production capacities, plant line selection for given product, decision making, planning and assessment related to the plant and process.

The problem that has to be solved is a combinatorial optimization problem. Even when the user selects to use short-cut models to adapt operation duration, non-linear (e.g. stepwise) functions are evaluated to compute the objective function, rendering this problem a non-linear integer system. Such systems are usually NP-complete or NP-hard[1] and are assumed not to be solvable in polynomial time. Various algorithms and methods have been developed to tackle similar problems that can be classified in three main categories: heuristics, mathematical programming and randomized search algorithms.

Integrated process development can be more rigorously defined as being optimal with regard to several factors (multi-objective optimum). The objectives considered are typically costs, productivity, energy consumption and environmental impact. In many cases, such multi-objective optimal designs target for example at reusing mass and energy from the output stream of one process in the input stream of another process within the boundaries of a given system (e.g. a building, a facility) – making mass and energy flows "integrated". In this context it is obvious that practical solutions have to be profitable. The challenge lies in finding process designs delivering high profits while having only little environmental impact thus being eco-efficient (Hungerbühler et al., 1999, Jankowitsch, 2000).

Problems arising in the design and operation of batch chemical plants can be described and solved by many different methods. In the 70's and 80's increasing trend of massive use of the computers allowed researchers to develop and revise computational methods in the field of searching for optima (Weisstein, 2006). As our problem is described as discrete, many commonly known techniques[2] of finding the extreme of a multi-parameter function are not applicable.

A heuristic method, tailored for a specific optimization problem, sacrifices the solution quality to enhance the computational feasibility. On the other hand, the class of rigorous methods is composed of several general equation-based mathematical techniques such as *Mathematical Programming* (LP, MILP, NLP, MINLP) and *Dynamic Programming*. These rigorous methods can guarantee the optimality of the solution in finite time for some problems (P and NP-complete Problems) (Garey and Johnson, 1979). However, for large size of NP and NP-hard class of problems, the rigorous methods are computationally infeasible. Numerous

1. In computational complexity theory, NP-hard (Non-deterministic Polynomial-time hard) refers to the class of decision problems that contains all problems H, such that for every decision problem L in NP there exists a polynomial-time many-one reduction to H, written $L \leq_p H$. Informally, this class can be described as containing the decision problems that are at least as hard as any problem in NP.

2. i.e. Nelder-Mead method, continuous steepest descend method, Newton method, Flatcher-Powell method, . . .

practical problems are mixed integer optimization problems, that are intractable. Such problems commonly are addressed with heuristics that provide a solution, but not information on the solution's quality.

There exist many discrete algorithms for tackling the problems of finding a global extreme, or an extreme of a nonlinear function in an NP-hard setting (NIST, 2006).

The global optimization techniques, for example mathematical programming methods, such as: linear programming (LP), mixed integer linear programming (MILP), mixed integer non-linear programming (MINLP) require exact stipulation of the batch design problem. The stipulation has to be explicitly formulated for each problem separately, including detailed specification of tasks to equipment assignments. The complexity of the batch design problem also leads to simplified mathematical models, which often require feasibility check for each solution and an exact definition of equality/inequality constraints (Grossmann et al., 1983, Voudouris and Grossmann, 1996, Papageorgaki and Reklaitis, 1990, Mockus and Reklaitis, 1999). From the broad range of mathematical programming methods, only few examples successfully handling variety of batch process design problems are mentioned below.
The linear programming (LP) (Yoo et al., 1999) was used for general retrofitting of a batch process. It is a simplification of the nonlinear mathematical programming.

Many authors use the Mixed Integer Nonlinear Programming (MINLP) methods with this type of problems, e. g. (van den Heever and Grossmann, 1999, Mockus and Reklaitis, 1999, Papageorgaki and Reklaitis, 1990). Branch and Bound Method (Gross and Roosen, 1998, Grossmann and Floudas, 1987, Patel et al., 1991, Sparrow et al., 1975) utilizes mathematical programming in a way where the solution domain is divided into subdomains by reference to integer variables and the subbranches are searched for better values than the current optima. After identifying which branches do not contain the global optimum, the domain space is reduced substantially.

Randomized heuristic and metaheuristic search algorithms provide an edge in the feasibility of solutions to many NP-hard problems compared to MINLP methods, but don't guarantee optimality of the solution. See for example (Zitzler et al., 2001, Michalewicz, 1994, Pham and Karaboga, 2000) for comprehensive overviews and applications of the randomized search methods.

Genetic algorithms (GA) offer advantages in the large scale problem and are inspired by nature's evolution strategy. The solutions are represented as codified genes. The genes contain all relevant information about the solution. Additional functions or operators are required to modify the solution genes such as: crossover, mutation, elimination of low performance solutions, learning, etc. In the field of chemical batch processing, the GA solves many problem types, although it is necessary to define controlling mechanisms in order to avoid invalid solutions to be generated (Chen et al., 2003). For the multipurpose batch plant optimization (Dietz et al., 2006) published a method for evaluating process designs according to multiple criteria including cost objective functions. The aim was to estimate the Pareto-front[3] by multiobjective genetic algorithm (MOGA). GA optimization framework inherents difficulties related to a large percentage of invalid or infeasible solutions in the population if random generation of chromosomes is used

(Areibi et al., 2001, Dietz et al., 2006). In the study by (Chen et al., 2003) a new application of GA has been conducted for a combinatorial follow-up design problem and several EA performances have been compared.

Tabu Search – a meta-heuristic described by (Glover and Laguna, 1997), has been used for solving different types of problems and provides a good compromise between computational time and results quality. Several practical implementations of Tabu Search algorithm have been published regarding batch plant processing optimization. In the article published by (Wang et al., 1999), a special Tabu Search (TS) optimization algorithm with double tenure list was used successfully and applied to the multipurpose batch plant design. Better results were obtained in comparison with the results of mathematical programming (MP) and simulated annealing (SA).

(Balasubramanian and Grossmann, 2003a) have also implemented a TS method for the scheduling of processing plants under uncertainty and compared it to a mathematical programming approach. They found TS to be an attractive method to obtain in a relatively short time good solutions for large problems that might be intractable with standard MINLP optimization techniques.

(Armentano and Arroyo, 2004) use multiobjective TS to obtain results in a bicriteria flowshop problem and demonstrate that the TS cannot only find the global optima in the defined problems, but also find in some cases better solutions than published before. (Lin et al., 2005, Lin and Miller, 2004) defined a methodology used to solve a wide variety of chemical engineering problems. They demonstrate on several small NLP and MINLP test cases and three small- to middle-scale chemical process synthesis problems the feasibility and effectiveness of the TS techniques.

(Cavin et al., 2004, Cavin et al., 2005, Mosat et al., 2005a) studied a multiobjective optimization of multipurpose batch plants using Tabu Search as a meta-heuristic applied to the problem of finding a set of "high-performance" process-layouts. TS makes use of adaptive memory to escape local minima. As shown in these publications, the Pareto-front approximation of solutions is often not a sufficient criterion for decision-making if the solution, a process layout, has to be implemented in real conditions. Therefore a methodology is presented, which allows for finding a non-dominated solution, Pareto optima and a diversified selection of feasible solutions to the batch process design problem in order to avoid infeasibility of addressed solutions due to insufficient complexity of the underlying mathematical model of the process. This methodology is similar to the *Rough Set Method* and *Ranked domain drift group analysis* presented for example by (Yanofsky et al., 2006), incorporating the knowledge of an expert within the optimization algorithm. The result of this process is a set of all possible candidates for the optimal solution, called the Pareto domain, that have been ranked according to the expert's preferences. In addition to providing the optimal solution to the problem

3. Pareto-front is a set of states of objective parameters satisfying the criterion of Pareto optimality. Pareto optimality is optimality criterion for optimization problems with multi-criteria objectives (multi-criteria optimization). A state A (a set of object parameters) is said to be Pareto optimal, if there is no other state B dominating the state A with respect to a set of objective functions. A state A dominates a state B, if A is better than B in at least one objective function and not worse with respect to all other objective functions

at hand, the ranked Pareto domain can yield useful information regarding the robustness of the optimal operating point.

1. 1.2 Probabilistic batch design problem

In the literature, the probabilistic batch design problem is mainly cited in the scope of scheduling and production requirement satisfaction (Grossmann et al., 1983, Reklaitis, 1995), where the allocation of tasks to units or vice-versa is considered under uncertain production variables or uncertain per annum production quotas for a number of products. Hence, uncertain production requirements are the basis of uncertain optimization, see i.e. (Shah and Pantelides, 1992) and the resulting multipurpose batch plant solutions are penalized for infeasible production regions.

Usually the production quota is given by customers and the delivery time is a function of the available plant equipment and equipment sizes, etc. At the end, the most important factor for delivery time is the productivity measure – typically the performance of design, which should be maximized. Maximizing of the productivity leads to inflexibility of a batch design, as shown by (Mulvey et al., 1995) and vulnerability of the design performance by change in operating conditions. Therefore a compromise between peak-performance, which is often the case in productivity-optimal designs, and invariance to uncertain operating conditions has to be achieved.

In the extensive review by (Sahinidis, 2004), probabilistic batch design problems solved by mathematical programming methods are enlisted. The authors recognize, among many sub-problems in the uncertain optimization of batch processes, that the most important topics are: *minimizing deviation from goals*, *robustness* as a quantified criterion of process quality and *flexibility* of a design. Mathematical methods are presented as solvers for such kind of problems. The principles of handling uncertainty in process development remain independent of the method used. One of the successful methods for handling uncertain models within the optimization is *programming with recourse*, a *two-stage* here and now programming, where the *first-stage* variables are those that have to be decided before the actual realization of the uncertain parameters. Subsequently, once the random events have presented themselves, further design or operational policy improvements can be made by selecting, at a certain cost, the values of the *second stage*, or *recourse*, variables. We use the approved *two-stage here-and-now programming* method in connection with a metaheuristic multiobjective TS algorithm for solving the multiobjective robust design problem. Before defining the actual robust design problem, we will review robustness and flexibility criteria in the literature.

Robustness of a batch design

In the literature, a robust batch process or the term "robustness" is in all cases related to uncertainty, but definitions of the term differ substantially. Basic distinction can be observed among the definitions: 1. ***flexible design problem*** and the 2. ***robust design problem***, which will be used for the purposes of this literature overview. Therefore the term *flexibility*, although in the literature often cited as robustness, will be used in connection with the first problem definition. The robustness criterion will be used in connection with the second problem definition.

1. ***The flexible design problem***.
   The ***flexibility*** criterion's goal is to provide quantitative measures of ensuring feasible performance of a design under all considered uncertain conditions. In this sense, a flexible design could be understood as a reliable design, when for example after one equipment unit fails, the design is still operable. This term is also known under "*probability of feasible operation*" (Straub and Grossmann, 1993).
   Some of the applications of the flexible design problem include:
   - quantitative criterion as an index for operational flexibility (Swaney and Grossmann, 1985).
   - flexibility as a process design reliability (Kubic and Stein, 1988), handles computations by fuzzy sets programming method.
   - stochastic geometric programming applied to engineering design under uncertainty by (Avriel and Wilde, 1969), where the flexibility is used as a combined criterion of satisfying the upper and lower bounds for operation variables and costs in the optimal condenser design problem. The method uses a mono-objective two-stage permanently feasible geometric programming approach.
   - flexibility of various problems related to design of a problem was examined by (Bansal et al., 2002), where parametric programming was used. The method is applicable to linear or nonlinear problems.
2. ***The robust design problem.***
   The robust design criterion's goal is to provide a quantitative measure of process stability under uncertain conditions so that the performance/profit variance is minimized. The robust design method aim is to identify a design in which the influence of all uncertain input parameters in all possible combinations is minimized.
   Robustness objective applications in the batch process design include:
   - (Bernardo et al., 2001) identified and incorporated quality costs and robustness criteria in chemical process design problems under uncertainty within a single-level stochastic optimization formulation. The solution defines an optimal design, together with a robust operating policy that maximizes average process performance.
   - (Ahmed and Sahinidis, 1998) defined robustness as a goal programming approach to balance trade-offs between expectation and variability of the recourse cost in the batch process development. The linearized form of a non-linear problem in a two-stage programming was applied.
   - (Nishida et al., 1974) defined robustness as a measure of attaining the min-max performance structure of a design, which should minimize the effect of the worst variations in the uncertain parameters. It uses mono-objective optimization.
   - A robustness measure in the monoobjective cost optimization of a chemical process with an objective function including error parameters as a result of uncertain inputs has been proposed by (Painton and Diwekar, 1995). The measure is quantified as a penalty to the objective function. Simulated annealing is used in this case.
   - (Samsatli et al., 1998) considered robustness as a measure for a reasonable performance over a wide range of uncertainty. They define some general robustness metrics that can represent significantly different robustness objectives simply by modifying functions and parameters.

- Robustness as an optimal design of systems involving parameter uncertainty characterized as a minimum average normalized deviation of the objective from the optima over the range of uncertainty (Wen and Chang, 1968). The proposed definition of robustness involves minimizing the expected relative sensitivity, which may be defined according to the probability distribution of the system parameter. The method is monoobjective.
- (Bonfill et al., 2005) addressed robustness in the scheduling problem with uncertain operation times

Another approach combines both flexibility and robustness into one criterion for obtaining both flexible and robust designs. (Rooney and Biegler, 2003) formulated the combination of feasibility problem (model parameter uncertainty) and robust design problem, where robustness is a quantifier of the operating parameters response.

(Mulvey et al., 1995) defined the term Robust Optimization (RO) and differentiated between: *model robustness*, which is important for mathematical programming models, where the authors state that robustness/feasibility of such model is usually overemphasized, and *solution robustness*, which is then particularly important in the context of optimization formulations. They also showed that robustness, as an objective function, is antagonistic to the costs-performance objective function. This principle will also be demonstrated on a case study in the Chapter 4 *Uncertainty with application to design robustness measures*.

The development of new processes has therefore gained in complexity, due to these new constraints or objectives. But the growing globalization simultaneously requires faster time-to-market and hence tends to reduce the time a company can invest for process design and optimization. In particular in fine chemistry, pharmaceuticals and custom manufacturing, chemical companies must continuously develop their product portfolio (and hence introduce new production processes) in order to maintain their competitiveness. Hence a reliable, efficient and rapid process synthesis – that takes into consideration all the objectives and constraints – is one of the keys for a successful business in these sectors.

1. 2 Research focus

In the traditional sense, batch process design is understood as a final part of Research & Development (R&D) for a given product. Figure 1-1 shows a scheme of the chemical process development in custom manufacturing, with an aim to create profit for a company by fullfilling the order of a chemical. In the first place, the customer is the initiator of such a request. The product specification in early phases can lie anywhere from underspecified (not even amount that will be ordered is known) to perfectly defined (probably in the case where the customer already ordered the same compound before). After the initial screening, request is usually formed from a customer order. At this point, in many cases neither the final selling price nor delivery date is known. If the compound has not been manufactured before, the production recipe is also unknown and has to be developed first in the laboratory scale. Usually process chemistry is not included in this first stage. After the laboratory recipe is defined, the bench- and pilot-scale experiments are performed. This is the first point in time when engineers start concerning about the process layout, that means how the chemical operations will be allocated in the factory. For reasons such as: due dates, legislation and similar, the production layout should be fixed as soon as possible. Therefore the most acute opportunities for design optimization, that means performance or cost optimization, arise early in the process. In the ideal state, the optimization begins directly after the request is formed, which is rather unrealizable.

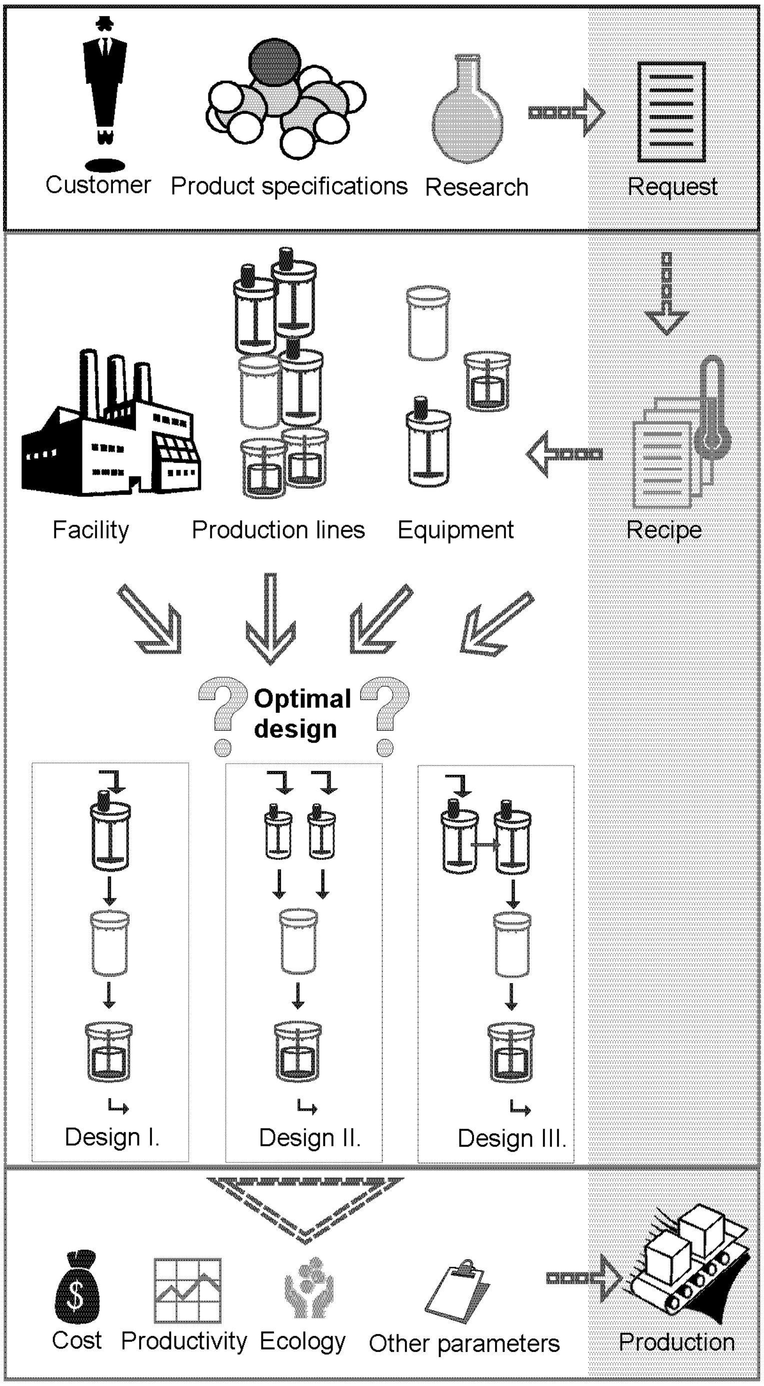


Figure 1-1: Overview of the batch process development in custom manufacturing from the order to the final production step.

Design optimization is performed according to multiple criteria, where the priorities of the objectives are known. Objectives are in most cases the costs and related productivity, ecology, safety, eventually others such as: simplicity of the process, robustness under varying conditions, etc. After the optimal design is identified, at present stage in industry, mostly by non-automatized calculations, the production phase can begin.

1.2.1 Objectives

A novel approach for solving various design problems related to single products in multipurpose batch plants is presented.

The main goals of the research posed in this thesis can be summarized as:

1. developing methodologies and algorithms supporting the batch plant engineering development team during the process design phase
2. finding and demonstrating a practical method of reducing significantly the size of the batch design optimization domain
3. stipulating and investigating a stochastic mathematical model related to chemical batch recipe which can be handled by multiobjective optimization in a reasonable time frame allowing rapid result output.

As a first step, the problem was defined as a single product multipurpose batch design optimization, where the single product is to be manufactured in a given plant. The method will be referred to as "the conventional TS optimization". In the following chapters, we extend the conventional TS method and include: new universal modular Tabu Search algorithm, new objective functions, a new approach solving the batch design problem by a "superequipment concept" and introducing uncertainty into the formulations.

The problem domain of early stage batch process development will be extended in the following way: in addition to the optimization of a single deterministic design, the new optimization algorithms assist in: retrofitting of equipment, grass-root design of a new plant, automated production plant line selection or in evaluating a design robustness by an automated stochastic method.

1.2.2 Automated deterministic batch process development and assessment

Here, the problems related to single deterministic design optimization are handled. A range of objective functions is defined, resulting in a results set comprising for instance: costs, diversification of resulting designs, simplicity of a design, NPV, ... The new objective functions aim at providing more detailed information about proposed design set to the decision maker. Currently 12 objective functions related to deterministic problems are programmed and up to 5 can be included in a prioritized list in a multiobjective optimization.

As the multiobjective optimization problem is nonlinear and NP-complete, an efficient solving algorithm has to be defined. In this thesis a Search algorithm is defined as an optimization basis and demonstrated on multipurpose batch plants process development problems.

1.2.3 Superequipment – an efficient model for reducing the optimization domain complexity

Problems related to the modification of a plant line by adding one or more equipment units, the selection of a plant line out of several available, or the definition of a whole new plant are handled. A new approach called *superequipment concept* has been developed for solving this set of problem types. For each problem type a different list of objective functions can be selected.

In the novel approach, the concept of superequipment is defined as an abstract model utilizing a virtual unit, which is capable of performing any physico–chemical batch operation. Each superequipment is transformed into a real equipment unit, for example a reactor, during or after the optimization in order to evaluate performance parameters of a design. This novel concept uses an implicit definition of a superstructure and essentially optimizes on the transfers between different equipment units in a design.

The superequipment model helps in determining the optimal equipment units for a given recipe and is applied on the following problem specifications:

1. Plant retrofit for a given plant line and single recipe.
   This problem includes investment in an existing plant line. The upper limit of investment in terms of number of equipment has to be set in order to obtain realistic designs and results. Net present value is usually set as the most important objective function in the prioritized list of objectives. The method identifies those equipment units that provide the highest Net Present Value (NPV) and determines corresponding process layouts. The equipment units are characterized by equipment class, size, lining material, additional options (attached condenser, distillation column) and costs.
2. Plant line selection for given recipe.
   The goal is to obtain a unified sorted list of optimal designs from all plant lines, plus the information about the investment scenarios into additional equipment if the capacity of a certain plant line is not sufficient. Usually there is a compromise between selecting a large plant line with high throughput versus a small plant line favouring the simplicity of designs. As the list of results stores diverse designs from each plant line, the method offers important information for decision making.
3. Grass root design for a single recipe.
   This problem demonstrates grass–root optimization of a batch plant, offering all the necessary details on each stored design, such as: size of the units, NPV, vessel options, lining materials, multiple choices for selected units and more. However this is limited to one product up to date, which means the case is constructed as monoproduct batch plant. The grass–root design case offers the testing of the concept using a larger number of superequipment.

1. 2.4 Stochastic nonlinear two-stage multiobjective optimization

The new method poses inclusion of uncertain variables into the multiobjective optimization algorithm. This concept is demonstrated on a single product to be manufactured in a single multipurpose production plant line under uncertain recipe variables. The uncertain recipe variables can be for instance: operation time and operation volume.
The response to uncertain operating conditions, as a quantitative measure of a batch design will be referred to as "*Robustness of a design*", or short: robustness.
As a *novel* technique, inclusion of performance robustness as an objective function alongside with productivity of a design results not only in optimal performance design set or solely robust designs, but both optimal performing and robust designs in one.

The application of all proposed multiobjective optimization methods is demonstrated by case studies.

1. 2.5 Overview of the thesis content

In the **Chapter 2**, the multiobjective optimization algorithms will be presented. This chapter comprises the basic TS formulation, the extension to the superequipment concept, the definition of the various objective functions, the approach for incorporating uncertainty and the implementation of the whole concept in terms of software.

On the basis of the mathematical formulations a set of case studies will demonstrate in **Chapter 3** the application of the superequipment concept. The case studies comprise: grass root design of a new batch plant, retrofitting of equipment units in a selected plant line and a selection of a suitable batch plant line for a given recipe.

A novel robustness measure of a design as an objective function in a multiobjective optimization will be demonstrated in **Chapter 4**. A case study involving uncertain recipe definition to be produced in a multipurpose batch plant is presented.

Chapter 5 presents conclusions and discusses possible future developments on the basis of the presented methods.

Methods & algorithms

2

Abstract

The aim of the method presented here is to optimize the design of a single chemical process to be implemented in an existing multipurpose batch plant. After giving a detailed problem definition the different steps of the procedure are explained in the following paragraphs. This chapter formulates the deterministic and stochastic batch design problems that are tackled in this thesis. A solution to these problems is a design, i.e. an allocation of recipe tasks to batch plant equipment. For this, a new approach introducing so-called superequipment is presented.

An improved modular Tabu Search algorithm will be outlined for handling multiobjective optimization problems. Objective functions as quantitative measures of the quality of process designs are mathematically expressed afterwards.

Applications and case studies are then presented in Chapters 3–4.

2. 1 Batch process design problem

After specifying the batch design problem we introduce the design criteria in form of objective functions. The definitions are generally valid for both deterministic and stochastic batch process design problems. If some definitions or equations are valid only for the stochastic problem, it will be stated specifically for such case.

2. 1.1 Problem definition

The presented approach is used for solving the following problem definitions:

Given:

Recipe

- expressed as a sequence of chemical/physical tasks
- capacity requirements for each task of recipe per unit of final product
- base duration of each task at the input scale
- recipe constraints (allowing or forbidding certain order and combinations of tasks)

Plant data

- equipment description including detailed specifications such as: nominal volume, operating T–P ranges, lining material, additional options (attached condensers, distillation columns), etc.
- connectivity constraints among equipment in the plant line

Economic data

- detailed cost composition on campaign basis
- investment costs where applicable

Heuristics

- which equipment class is capable of performing which recipe operation classes
- design heuristics
- scale–up rules for each operation class expressed as a function of batch size and equipment class
- heuristics for the optimization related to superequipment (e.g. determining the equipment class for superequipment)

Optimization parameters

- one or more objective functions

Determine:

A set of dominating (approximated Pareto–optimal) plus structurally diverse dominated layouts for the process, i.e. allocation of recipe tasks to equipment units, structure and order of the final recipe (e.g. in parallel or in series use of units)

2. 1.2 Overview of the method

A general overview of the method is presented in Figure 2-1. The data inputs represented as cylinders plus the constraints are needed for the design optimization. The process routines for recipe analysis, superstructure generation, process simulation, and optimization have been implemented in a Matlab® program. The process simulations can also be conducted in a commercial batch process simulator.

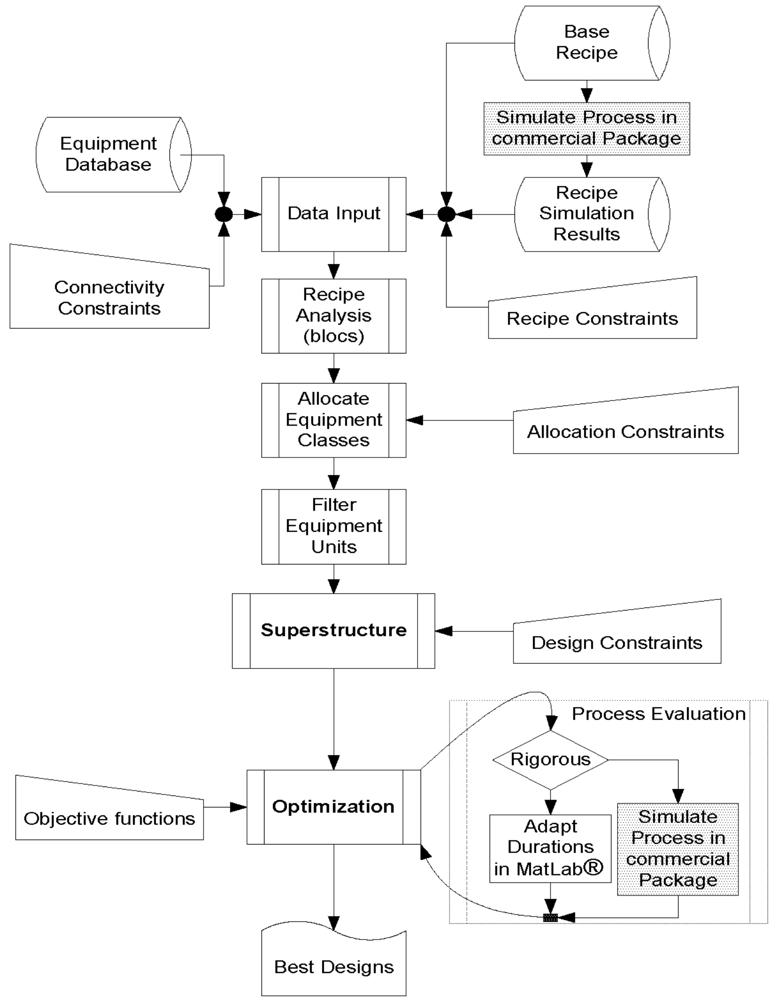


Figure 2-1: Overview of the batch process design method. The gray rectangles mark usage of optional external software, the rectangles with parallel vertical lines indicate the main process data flow (Cavin, 2003).

2. 1.3 Inputs and term definitions

This section describes in detail each input component from the Section 2. 1.1 *Problem definition*. The list of corresponding mathematical operators is given in Table 2-1.

Table 2-1: List of operators. The letters used in the definitions are variables.

$A \leftarrow B$	B is transformed to A
$A \in B$	A is contained in B
$A \notin B$	A is not contained in B
$A \subset B$	A is contained in B (subset, group operator)
$A \supset B$	A contains B (group operator)
$A \cup B$	union of A and B
$A \Leftarrow B$	B is assigned to A
$A.B$	column B of matrix A
$A \mid A.C = d$	take rows of A where the value of column C is d
$A.B \mid A.C = d$	take only column B (same rows as above)
$\forall A : B$	for all A the dictum B is valid
$A \oplus B$	pair-wise combination of A and B
$A \neq B$	A is not equal to B
$A \cap B$	intersection of A and B
$\bigcup_{i=1}^{n} A$	union of A over i=1..n
$\{A\}$	set containing A
$\varnothing$	empty set

2. 1.3.1 Recipe

The *Recipe* [*R*] is an ordered sequence of physico-chemical tasks with detailed task specifications. *R* is a matrix containing vertically the physico-chemical task rows. The vertical position, also called index *[R.ID]*, represents the position of the given task in the sequence of tasks (see Figure 2-2a).

Each row of *R* first indicates the type of the task [*R.OpClassID*], the base volume [*R.Volume*] and the estimated base duration [*R.Time*] required for the task. All column names ending in ID are relations to lists of available options stored in library matrices. For instance, *R.OpClassID* refers to the operation class matrix [*OpClass*] (or in a short form [*oc*]) containing all supported types of tasks (e.g. reaction, distillation…), see Figure 2-2c and Table A-2. Operating temperature [*R.Temperature*], pressure [*R.Pressure*] and required lining materials [*R.LiningID*] are then given. Each step also contains additional information in form of *"flags"* [*R.Flag*] that encode accepting or refusing parallel or serial use of units for processing, required, allowed or forbidden transfers during or after the block. Based on *R.Flag*, where one flag indicates the impossibility of a transfer between two subsequent tasks – i.e. the two tasks must be conducted in the same equipment unit(s) – the recipe is condensed in a block matrix *B* (see Figure 2-2ab). Each no-transfer block row of the B matrix contains a link to the corresponding recipe R rows [*b*]. Each row of *B* contains the largest volume required during the block, as well as the highest pressure and temperature reached (the maximal values for each block operation are marked in bold case on Figure 2-2b). The block [k_i] from the bolck recipe matrix *B* is defined by Eq. 1.

$$k_i := R \mid R.ID = b \qquad (1)$$

The recipe also includes information about the base case batch size, which is computed from the material balance of the basic design given as an input. The material balance computations are performed by external software tools or are to be entered directly to the optimization algorithm. Base case in this scope is the state for which the material balances have been computed and refer to the basic layout of the process, usually the simplest possible arrangement of equipment to perform all the tasks of the recipe.

Per definition, any variable in the recipe can be considered as uncertain. In the following sections, we define as uncertain the time and volume requirement of individual tasks, which results in the uncertainty in cycle time (CT) and batch size (BS) for each generated design.

(a) **[R] Recipe matrix:**

R.ID	R.OpClass	R.Time [min]	R.Volume [m3]	R.OpClassID	R.Flag	R.Temperature [°C]	R.Pressure [bar]	R.Lining	R.PrevOp	R.NextOp
1	Charge	20	8.4	4	0	20	1	-1	---	2
2	React	230	8.4	32	1	180	4	-1	1	3
3	Charge	20	9.4	4	0	90	1	-1	2	4
4	Distill	160	9.4	17	2	80	0.5	-1	3	5
...	...	...	...	...	...	...	...	...	...	...

(b) **[B] Block Recipe matrix:**

BlockID (k_i)	B.OpClass	B.Time [min]	B.Volume [m3]	B.OpClassID	B.Flag	B.Temperature [°C]	B.Pressure [bar]	B.Lining	b
k1	Charge, React	20, 230	8.4	4, 32	1	180	4	-1	1,2
k2	Charge, Distill	20, 160	9.4	4, 17	2	90	1	-1	3, 4
...	...	...	...	...	...	...	...	...	...

(c) **[OpClass] Operation Class matrix:**

OpClassID	OpClass.Name	OpClass.Scaling
32	React	special scaling
4	Charge	constant scaling
...	...	...

(d) **[EqClass] Equipment Class matrix:**

EqClassID	EqClass.Name
14	Reactor
...	...

(e) **[A] Class Assignment matrix**

A.OpClassID	A.EqClassID
32	14
4	14
...	...

Figure 2-2: Input matrices of the batch process design problem (R, B, OpClass, EqClass). The R.Lining material requirement -1 denotes "any lining possible" for the particular operation.

2. 1.3.2 Plant data

Plant data is expressed as follows: *Equipment list* [*E*] is a list of all equipment physically present in a given plant. Each equipment is characterized by its properties: 1. equipment class, 2. unit volume, 3. construction material, 4. lining material, 5. attached optional components, 6. location in the plant. For investment scenarios, additional equipment units and information are available (see below).

Limiting operation conditions of each equipment unit are stored as a link to matrix [*P*], containing a list of standard temperature and pressure ranges (*TP ranges*) of units typically used in industry.

Operation class [OpClass] is a matrix containing: operation names, operation identification number (ID) and corresponding operation scaling rules (constant, linear, non-linear smooth and step-wise scaling of volume or time). For example: operation class: 'multidrop centrifuge', scaling of operation time according to volume: stepwise[4] (see Figure 2-2c and Table A-2)

Equipment class [*EqClass*] is a matrix containing equipment class name and equipment class number [*EqClassID*]. For example: equipment class reactor: 14 (see Figure 2-2d and Table A-1).

Operating conditions for equipment units pressure and temperature ranges of equipment are divided into temperature and pressure ranges, which correspond to industry norms for vessels. For example: TPrange 1: 0 – 280 °C, 0 – 6 bar, TPrange 2: -100 – 50 °C, 0 – 6 bar. The following lining materials are defined: 1, Stainless Steel/V4A (material standard); 2, Glass/Email/Graphite; 3, Hastelloy (metal alloy); 4, DIN 1.4539 (German industry norm); 5, PTFE. The lining materials are in the ascending order of resistance to corrosive or hazardous materials. If the lining material is not relevant for the recipe task (e. g. non-corrosive operation), a lining material notation "-1" is used consistently in this publication. In such case the lining material selection is not a limiting factor for the equipment selection.

Plant line equipment [*E*] is a matrix containing set of specific equipment units physically present in the given batch plant. Each equipment is characterized by its identification number, name, nominal volume, lining material, temperature and pressure range (TPrange as a part of *[P]*), floor in the plant. For example: ID: 1, unit name: 'Reactor 12', nominal volume: 16 m^3, lining material: PTFE,TPrange: 0 – 280 °C, 0 – 6 bar.

2. 1.3.3 Economic data

Economic data input is needed for operating costs (including material costs, waste disposal costs, energy, utilities, labour and other costs) and investment cost (e.g. when implementing new equipment in an existing plant line or grass root design for given recipe).

The economic data cost structure for a project has been implemented as a flexible data structure. The main cost composition components for a general batch process are listed in Figure 2-3. This data structure is flexible, that means every field can be renamed, modified, etc. This flexibility is achieved by an eXtended Markup Language (XML) implementation (Marchal, 2003) of the database. For the purposes of case studies listed in this thesis, the cost computation schema listed on Figure 2-3 is used, where the investment costs belong to the *other/user defined fields*.

---

4. Usually multi-drop centrifuge equipment is used for the operation class: multidrop centrifuge, in which case a certain fraction of the total amount of crystals is processed in a predefined period of time. The next charge requires again a certain period of time plus time for loading/unloading of materials.

A *List of equipment units* $[E_{buy}]$ that could be added to an existing plant, including the maximum allowed number of each class type serves as a basis for investment retrofit and grassroot scenarios. For instance a possible specification might be: at maximum two reactors and one multi–drop centrifuge can be added to a given plant, not exceeding two equipment units in total (e.g. due to space limitations in an existing plant). The investment related data for the retrofit scenario and grass–root design are computed according to "six–tenths" rule (Perry and Green, 1997) or searched in databases of existing prices (Matches, 2006, Cowan, 2003), scaled–up and corrected according to material, size, options and installation factors. The Net present value is computed according to cash flows specific for each campaign and design.

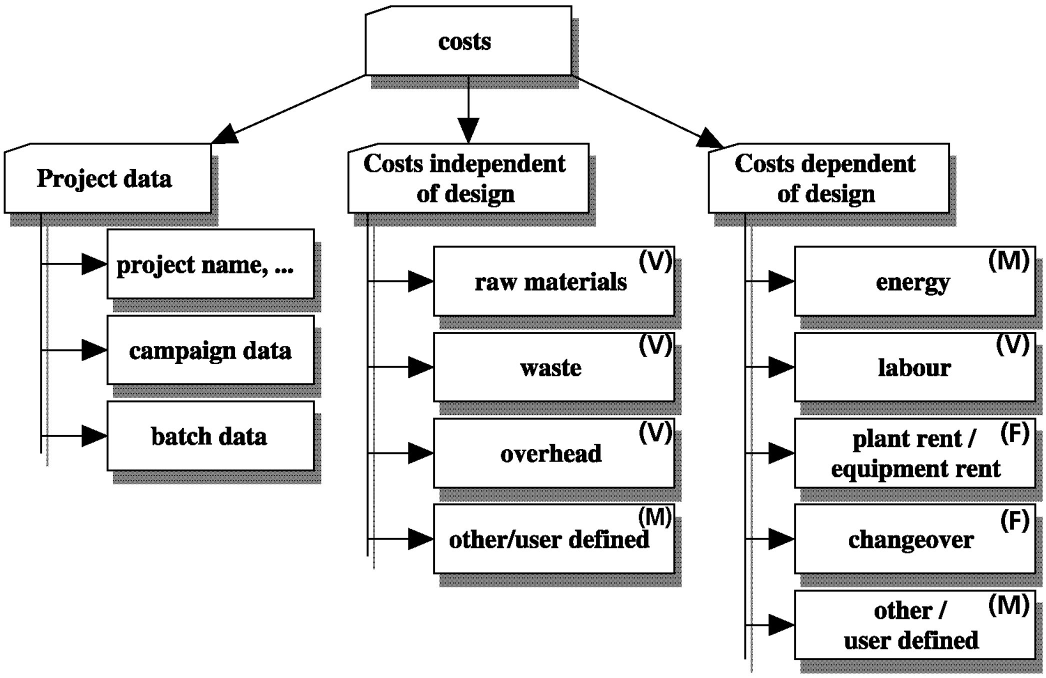


Figure 2-3: Project costs scheme as programmed in a XML database. Fixed costs are denoted (F), variable costs (V) and mixed costs (M) consisting of both the fixed and variable part. The categorization is specific for the presented case studies and might differ to industrial practice in some details.

2. 1.3.4 Problem related heuristics

Heuristics on implementing the design into the real facility include *assignments of given operations* to an equipment class in an assignment matrix [A].

Class assignments [A] is an assignment matrix of the type: operation class ID can be processed by a specified equipment class ID, i. e. the assignments contain multiple links of operations to be conducted in a given equipment class. For example: operation class 'reaction' with ID = 32 can be processed by equipment class 'reactor' ID = 14 (see Figure 2-2e and Table A-3). For instance reaction, crystallization, extraction and other operations can occur in a batch reactor, filtration, cake–washing and drying can be performed in a nutsche–dryer.

Additional heuristics are defined in order to ensure feasible and realistic designs in the results list, for example at maximum two units can be used in parallel, equipment cannot be used both in serial and parallel mode at the same time. These rules are implemented in the algorithm and applied automatically by the move definition in Tabu Search (TS) (see also (Cavin et al., 2004)).

2. 2 Tabu Search

The Tabu Search (TS) method (introduced by (Glover and Laguna, 1997)) is used to approach the investigated batch design problems. This algorithm is a metaheuristic global search method, and hence needs a stopping criterion. The heuristic component in the optimization has been selected after careful consideration of various aspects. The large size of the combinatorial problems in chemical batch processing applications often ensues NP-complete model formulations. The papers published by (Cavin et al., 2004, Cavin et al., 2005) discussed selection of proper optimization algorithm and show advantages of using Tabu Search algorithm for batch design problems. They also discuss parameter settings of such TS implementations, such as: neighbourhood size correlated to the input data, tabu list length, restarting options, aspiration criteria settings, etc. The TS implementation method presented in this chapter provides a wide selection range of batch designs, which include dominating (approximated Pareto–optimal) designs as well as dominated set of designs, in order to minimize the possibility that the majority or all of the optimal results are not feasible because of hidden constraints and limitations. Furthermore, designs are stored that are structurally different than the dominating solutions. Later in the text, we refer to the list of such dominating, dominated and structurally different solutions as the *results list*. The results list contains also additional detailed information on each design. This section will summarize terms, definitions and equations needed for the scope of explaining the optimization algorithms for various problem settings:

1. the deterministic batch design problem optimization
2. the superequipment concept
3. the stochastic batch design problem optimization

2. 2.1 General Tabu Search algorithm

This section details a general TS method, which is later used in combination with different problem specific functions to solve the problems introduced in Chapter 1 - Introduction.

Initial Solution

First an initial solution has to be provided; it should be as good as possible. If the initial solution is near the optimum, the number of moves needed to reach the optimum will be small, and hence the algorithm will find it quickly. Therefore, an initial solution proposed by experts or generated by heuristics is usually advantageous.

A method providing multiple initial solutions can however be preferable: the constraints may make the solution space non-convex, and hence a "good" initial solution may be computationaly very far (i.e. numerous moves) from the optimum. An efficient way of exploring the whole solution space is to restart with different randomly generated initial solutions. Further refinements can be implemented. For instance, a strategy used by (Wang et al., 1999) is to influence the random selection of the initial solution to cover previously unvisited regions of the optimization space. This allows a targeted diversification of the search.

Move Definition – Neighborhood generation

The definition of the moves, i.e. the definition of the modifications that can be done at each iteration to the current solution, is highly problem-specific. The current solution combined with the moves defines a neighbourhood. In general terms, with more moves allowed, the algorithm can be quicker (less moves needed to go to the optimum solution), and less prone to be blocked by constraints. However, as only a subset of the neighbourhood will be tested (see below), chances are also higher that the optimum will be missed. This is the first trade-off in the parametrization of the TS; as will be explained in the following, such trade-offs arise for most settings of the TS.

Neighborhood – Candidate list selection

According to the number and kind of moves defined, the neighbourhood can be quite large. If at each iteration every neighbour must be evaluated, the algorithm will become quite slow. Therefore, usually only a subset of the neighbourhood is considered. The most used method is to randomly select a fixed number of neighbours for consideration. The trade-off is then: the larger the sub-set, the slower the algorithm; but the smaller the subset, the higher the risk not to find the optimum – moves towards the optimum might simply not be selected in the subset of the neighbourhood.

Heuristics can be developed to either adapt the subset size (larger subset in unknown regions in order not to miss interesting solutions, smaller subset to escape a local minimum quicker), or to favour some "promising moves" over regular moves. These techniques, known as intensification, depend usually on knowledge gained during the optimization.

Tabu list

A special selection rule is set up to avoid loops. The latest move is placed on a tabu list, and the reverse move is forbidden during a certain number of iterations. This will allow the algorithm to escape local optima, and hence to conduct a global optimization. Indeed, when the optimum is found, the best neighbour will be an uphill move. In most cases, at the next iteration, the best neighbour will be the minimum just left; but as the reverse of an accepted move is tabu, returning to the previous solution is forbidden and the algorithm has to continue going uphill, i.e. leaving the local optimum.

An important parameter is the tabu list size, which indicates how long the reverse of one applied move will remain prohibited, as well as the number of moves that are forbidden (the list being usually managed as a First-In First-Out (FIFO) stack). The longer the tabu list, the smaller are the chances that the algorithm will loop around a local optimum. But the longer the list, the more limited the search becomes (good solutions could be missed because a move leading to them remained tabu for a long time).

A common optional improvement to the tabu management is aspiration. In order to lessen the limitations arising from long tabu lists, tabu moves may be accepted if they lead to a solution that is better than any solution found so far.

Objective function – best candidate selection

After the neighbourhood has been filtered to eliminate tabu moves and a subset has been selected, each neighbour in the subset is evaluated with regards to the objective function(s). The best neighbour is selected, and becomes the “initial” solution for the next iteration.

The objective function value (the “fitness” of a solution) can however be manipulated before the selection. In highly constrained problems for instance, constraints could be handled as penalties (and hence “validate” the solutions, but assigning them a lower value). The penalties should be large enough to promote valid solutions, and should be proportional to the number of constraint violations. However, they should not be so large as to effectively forbid such moves. The fitness can also possibly be modified according to other heuristics, in order to favour (or penalize) a “direction” – for example to favour exploring unvisited areas of the solution space. This method known as diversification allows a better covering of the overall solution space. It will depend on the regions visited during the optimization. As indicated above, this objective can also be achieved with multiple restarts in different regions of the solution space.

In the Figure 2-4 an algorithm scheme of a problem independent TS algorithm is defined. By detaching the data-flow from the TS algorithm, a certain degree of TS algorithm independence from the problem type was achieved.

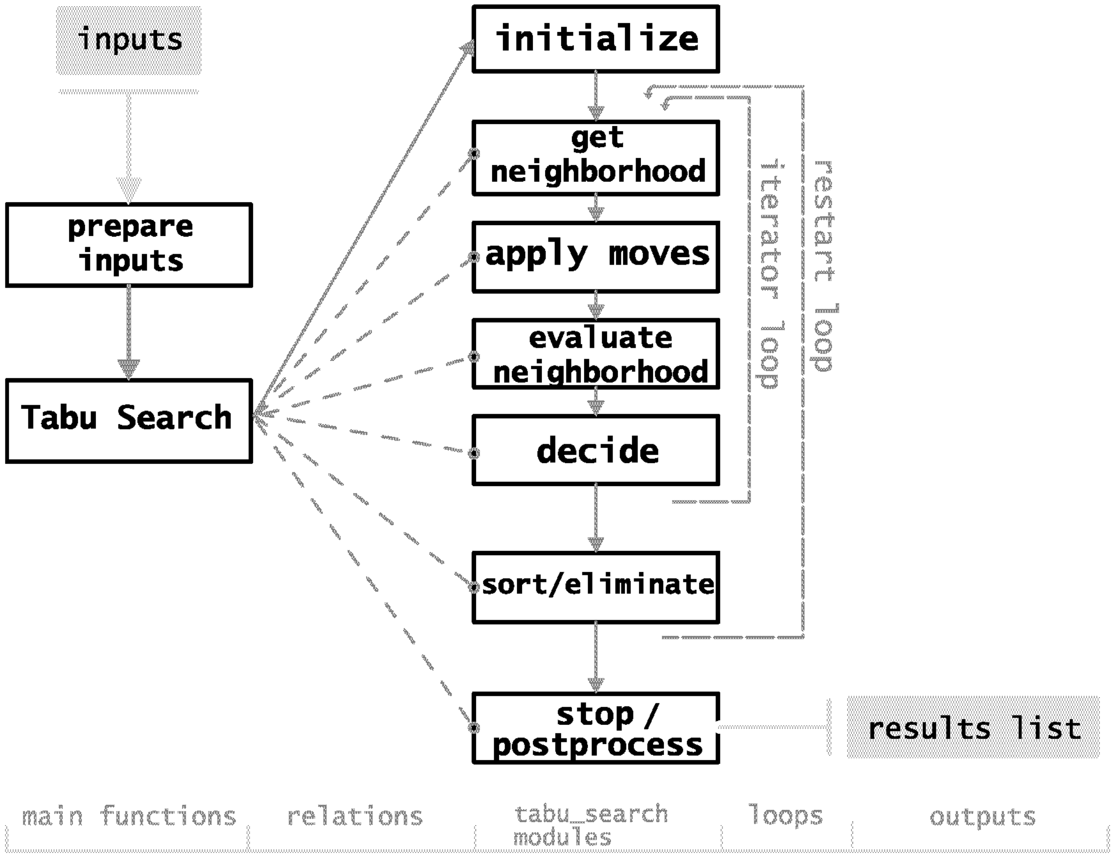


Figure 2-4: Modular and universal Tabu Search algorithm structure. The filled-boxes represent data storage, the full arrows logical data flow, the dashed lines hierarchy of the functions. In the program implementation, each non-filled rectangle represents a separate function.

The Inputs of the problem have to be defined centrally, processed by the *prepare inputs* algorithm and passed to the main function: *Tabu Search*. The Tabu Search function calls then all the problem specific subroutines, which will initialize the data, define neighbourhood, apply moves on selected solution candidates and evaluate the modified candidates according to multiple objective functions. The *decide* module returns the best candidate according to: tabu tenure, stored optima and advanced TS options. After the inner iteration loop has passed, the sorting/eliminating of solutions takes place. The restart loop refers to the optional restarting of the algorithm at random points. Stopping criteria have to be met in order to obtain the final optimization results. Usually the postprocessing of the candidates is the final phase in the optimization.

2. 2.2 Tabu Search applied to the deterministic batch design problem

The method aims at finding the optimal assignment of recipe blocks into given equipment units (Mosat et al., 2004). The transfers between the units are also determined and define the *moves* of Tabu Search. As an input, a base case layout (a design) and initial batch size, cycle time, operation durations, temperatures, pressures and other data are required. During each iteration one design is altered and all neighbours originating from that particular design are evaluated for their parameters (see Figure 2-5). The design parameters, such as task durations, volume and time requirements for each block are adjusted and scaled accordingly.

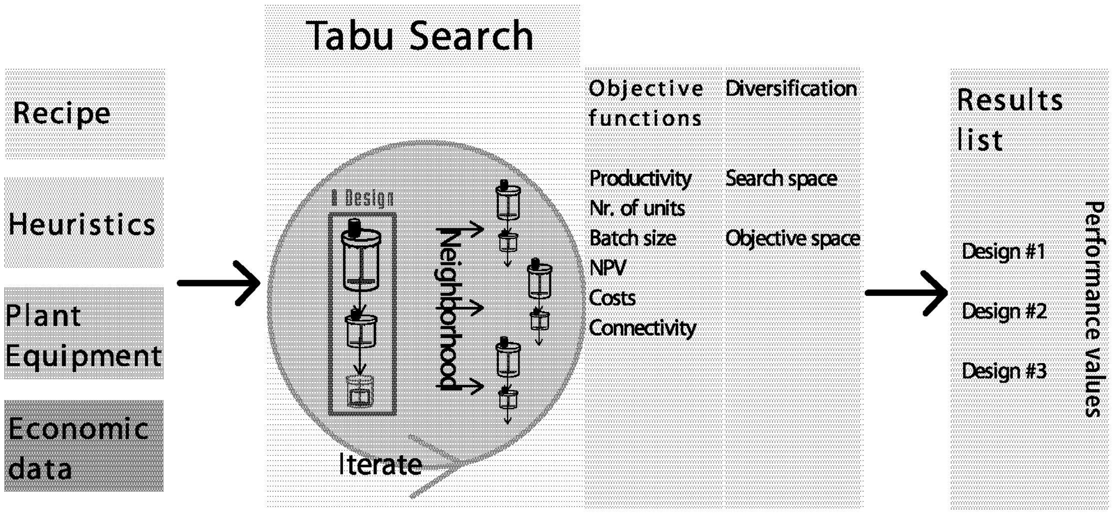


Figure 2-5: An overview of the implementation scheme of Tabu Search algorithm used for deterministic batch process design optimization.

The complete optimization algorithm was programmed as a software package in the Matlab programming language. In addition to the Matlab algorithms used for the TS program, additional software was used for input data definition (see Figure 2-6; Batch Plus™ engineering software for generating the plant line database [*E*], XML for defining the cost of campaigns and economic data) and as the output, the result reports are generated in webpages (HTML).

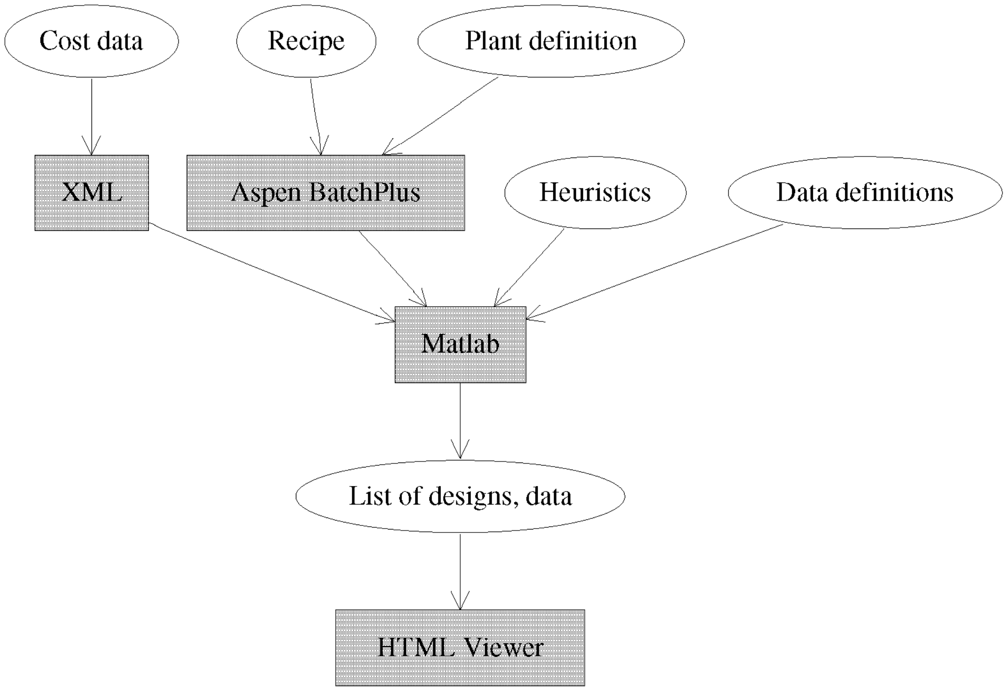


Figure 2-6: Software packages (grey boxes) used in different stages of batch process optimization for data processing (ellipses).

2. 2.2.1 Design – solution to the batch design problem

Design (L) is a solution to the multiobjective optimization problem representing a process flowsheet. Each design is defined as a set of assignments of given recipe tasks to a specific equipment unit. First we define the mathematical formulation and later a design example will be listed.

Let i be the main counter among all operation blocks in the recipe matrix B *(see Figure 2-2)*. For each block k_i from the block recipe matrix B feasible equipment classes $[U]$ are determined:

$$U_i := (\, A.EqClassID \mid A.OpClassID \supset (\cup\, k_i\, .OpClassID\,)\,) \tag{2}$$

Eligible equipment units from the matrix E (database of all equipment in the plant line) are stored in matrix V as defined by Eq. 3:

$$V_i := (\, E \mid E.EqClassID \in U_i\,) \tag{3}$$

Recipe block operations are defined as:

$$X_i := \{\, k_j . b_j\,\} \tag{4}$$

where the index j means that several subsequent blocks k can be grouped into one common composite block.

The next step in determining the design is to filter eligible equipment units V_i according to lining material and operating condition requirements as defined in Eqs. 5–9:

$$V_i \leftarrow [V_i \,|\, (P.P_{min} \,|\, P.TPrangeID = V_i\,.TPrange) \leq min\,(X_i.Pressure)] \tag{5}$$

$$V_i \leftarrow [V_i \,|\, (P.P_{max} \,|\, P.TPrangeID = V_i\,.TPrange) \geq max\,(X_i.Pressure)] \tag{6}$$

$$V_i \leftarrow [V_i \,|\, (P.T_{min} \,|\, P.TPrangeID = V_i\,.TPrange) \leq min\,(X_i.Temperature)] \tag{7}$$

$$V_i \leftarrow [V_i \,|\, (P.T_{max} \,|\, P.TPrangeID = V_i\,.TPrange) \geq max\,(X_i.Temperature)] \tag{8}$$

$$V_i \leftarrow (V_i \,|V_i\,.LiningID \in \bigcap X_i.LiningID) \tag{9}$$

The set L is a design which contains assignment of recipe blocks k_i into equipment units V_i:

$$L := \bigcup_{i=1}^{n} (V_i \Leftarrow k_i) \tag{10}$$

In the TS implementation we differentiate the units that are used in the design L, units available for moves (V) and units which are not available for the next move. Thus for each L_i there are applicable units stored in the vector (V_i).

The superstructure of each iteration step is defined by combining eligible equipment units for each recipe block with design rules and constraints. The first constraint is that no equipment unit can be reused, i.e. once emptied, it may not be used again in the same batch. Hence, if a given unit is assigned to a block, it cannot be used in another block, except if this would result in a continuous utilization. The units from previous block L_{i-1} and next block L_{i+1} can generally be used in the current block, if the condition of continuous utilization is maintained (Eq. 11):

$$V_i \leftarrow V_i - \sum_j L\,.E_{j \neq (i-1,\, i+1)} \tag{11}$$

For each design element L_i the condition of eligible equipment in L_i must be maintained. Finally one or several elements from the eligible equipment units are selected.

From design L we can obtain additional information from related matrices (k_i and E_i) such as: equipment size, lining material, floor in the building, mode of the operation (in series, in parallel) etc.

In the practical implementation, the design matrix L contains as many rows as there are blocks k_i in the block recipe B. The following modes of operations are implemented:

1. normal mode of operation, where one unit performs one or more blocks k_i
2. in parallel mode of operation, where two units process the given block(s) in parallel time frame

3. two units in series, where two equipment units are sequentially processing one block of operations k_i
4. three units in series, processing the same operation block k_i sequentially in time.

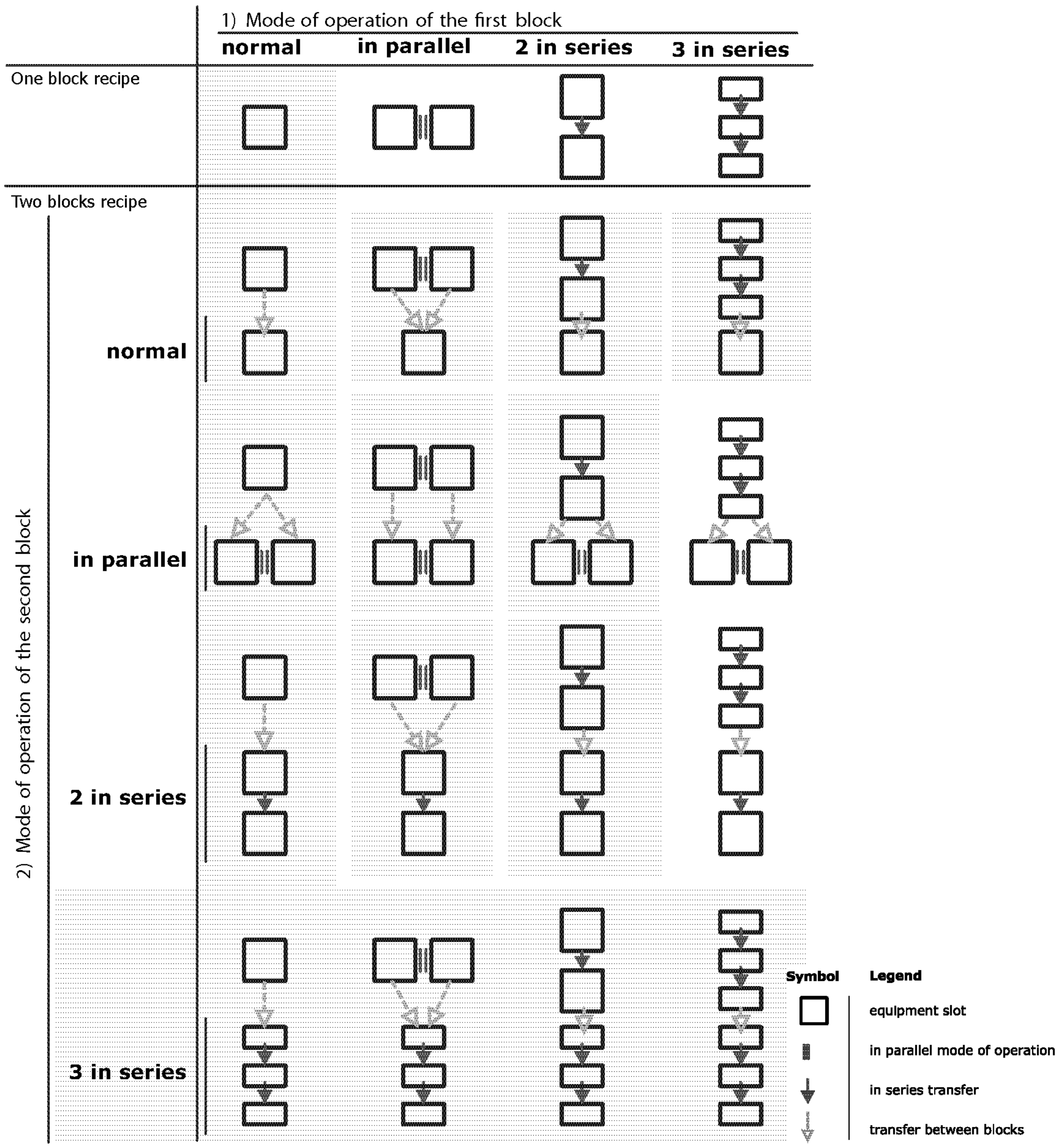


Figure 2-7: Design block combinations for recipe with one and two blocks. A rectangle corresponds to one equipment slot, full arrows to in-series transfer, two parallel lines to in parallel mode of operation and hollow arrows to the transfer from one recipe block to another. Solutions to recipes with more blocks are designed analogously.

If one recipe block is present, four operation mode combinations for a design arise (see Figure 2-7). For two recipe blocks, there exist 16 operation mode combinations. This number of combinations is only valid, if the recipe flags $R.Flag$ allow all modes of operations for the given block k_i.

An example on Figure 2-8 shows two possible arrangements of designs for two recipe blocks. The first design is valid according to the constraint of not refilling a once emptied unit. The second design would violate such constraint, because the first block in series will result in an empty unit A and the second block in series will result in re-filled unit A, which could introduce safety or quality related problems.

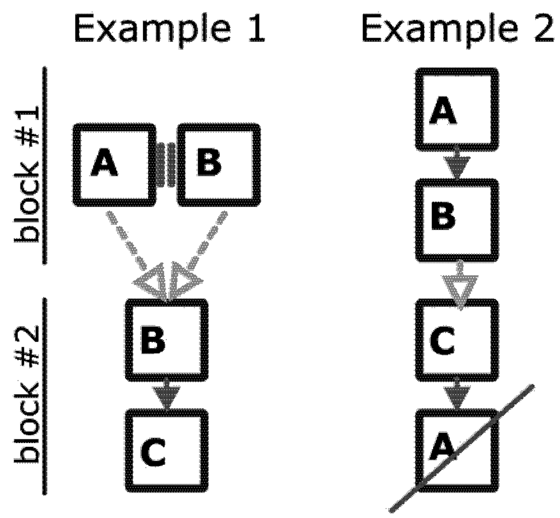


Figure 2-8: Equipment assignment to a specific slot in design examples. *Example 1* shows that the equipment unit *B* can occupy one slot in a parallel block #1 as well as one slot in the in series block #2. Example 2 shows that the equipment unit A cannot be assigned to the *in series* block #2, because this violates the "no reuse of equipment" constraint.

2. 2.2.2 Objective functions for multiobjective optimization of batch processes

As an input for the algorithm, a base case layout and initial batch size, cycle time, operation durations, temperatures, pressures and other data are required. During each iteration one design is altered and all neighbours originating from that particular design are evaluated for their parameters. The design parameters, such as task durations, volume and time requirements for each block are adjusted and scaled accordingly.

The algorithm has been designed for handling multiple objectives, where prioritized optimization objectives can be selected from the following list:

1. Production rate [*kg/hr*], maximize, the production rate is computed form the cycle time and batch size of the process designs.
2. Number of equipment units in design [*pcs.*], minimize, this criterion refers to the simplicity of design, from the practical point of view, the lower the number of units used in a design, the lower the maintenance, cleaning requirements, changeover and similar.
3. Batch size [kg], maximize, is the amount of product delivered by a single batch.
4. Number of significant equipment units including reactors, centrifuges, etc. [*pcs.*], minimize, as the important equipment units are sometimes used for tasks, which can be performed by other equipment, this objective function prevents such assignments. For instance, distillation can be performed by a separate dis-

tillation column or by a reactor with an attached condenser, in which case the distillation column is a non-significant equipment and the objective favors leaving the reactor free for other tasks.

5. Productivity per total nominal volume of significant equipment units used in a design $\left[\frac{kg}{h.m^3}\right]$, maximize, this objective function's aim is to maximize the volume utilization of significant equipment. It is important if the plant is in the so called Equipment Pool Plant (EPP) mode, where more products are manufactured concurrently in the same plant line and therefore optimization of the utilization has higher priority than a simple productivity maximization.
6. Floors–up indicator as a measure of upwards material transfer across the floors of a plant line [-], minimize, this function quantifies the pumping requirements in a plant, where usually designs with materials flowing from the top building floors to lower floors are preferred. This is used mostly as a less important objective.
7. Connectivity costs (or connectivity constraints violations) which prevent using impossible connections among the units in a resulting design and favour existing pipe connections between units [-], minimize. The connectivity constraints bidirectional matrix contains as many rows/columns as there is equipment units in the plant. Each tuple is refering to a connection from first equipment to second equipment on a scale from 0 - existing connection to 10 - impossible connection (if there is a thick wall in between).
8. Special function for obtaining diversified designs with regard to number of equipment units [-], maximize. The special diversification function serves as a technique for enforcing the searching of performing designs over a range of a minimal possible number of equipment units to a maximal possible number of units (which is given by recipe constraints and number of blocks). Maximal score of 1000 is set for those designs with highest productivity for a given number of equipment. If a design with the same number of units and better productivity is found, its score is set to 1000 points and the other designs' score is lowered by one. This is an objective space diversification method.
9. Special objective function for multiple plants selection [-], maximize. The special objective function for multiple plants selection is an index score and ensures storing of superequipment designs which fit into a selected existing plant line and lowers the score for designs which do not fit. See Section 2. 2.3 *Superequipment concept* for explanation.
10. Costs of campaign [*CHF, USD*], minimize, this objective function is explained below.
11. Net present value (NPV) of a project with or without investment [*CHF, USD*], maximize, this objective function is explained below.
12. Payback period [*years*], minimize, this objective function is explained below.

Costs of campaign

Costs of campaign (*Objective function* 10) serve as a basis for economic calculations and include cost of materials, waste management, labour, utilities, overhead, changeover etc. Before the production phase of a proposed batch design can take place, it is helpful and moreover necessary to estimate the costs of the process. Due to the nature of production line factory cost composition, the biggest contribution to the production costs are the raw material costs and the rent of the plant.

Figure 2-9 shows the cost objective function as a function of design parameters (batch param). The cost is a sum of costs related to: function of time ($f_{cost}(t)$), function of number of equipment used in design ($f_{cost}(nr.eq.)$), function of connectivity violations ($f_{cost}(v_{equip})$), volume of equipment used in a design and other costs ($f_{cost}(other)$).

For a case study, and better explanation of elements, please see the attached cost datasheet in Appendix A-2 *XML data for the cost function*.

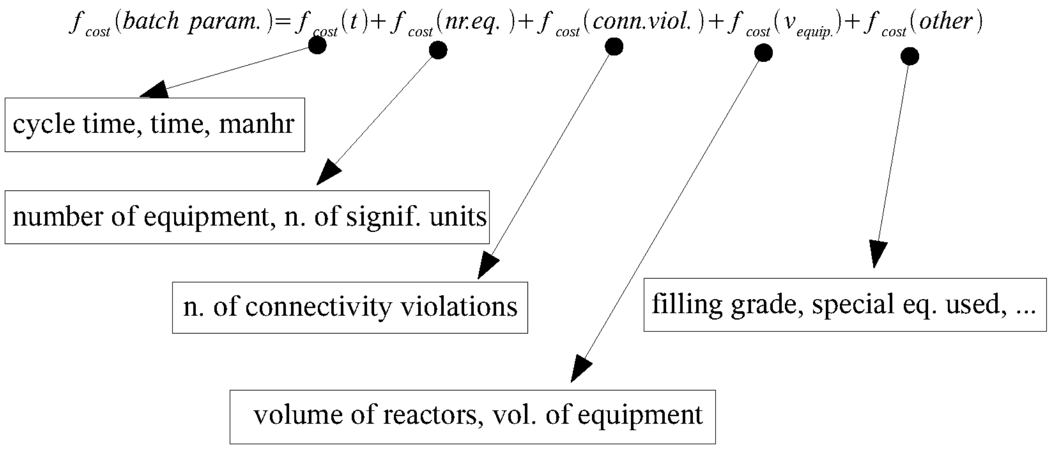


Figure 2-9: Cost objective function as a function of batch design parameters.

NPV

The net present value (NPV, objective function 11) is computed according to (Brealey and Myers, 1996) and, if applicable, includes investment. The interest rate used for the case studies from Chapters 3 – 4 is 10% and the time frame is scaled according to the length of the campaign.

Payback period

Payback period (*objective function* 12) is computed according to (Brealey and Myers, 1996) as the number of years before cumulative forecasted cash flows equals initial investment.

The objective functions are prioritized in an optimization scenario. Up to five objective functions can be selected for an optimization run. All selected objective functions are evaluated (see Figure 2-10) for each design. The objective functions' priorities are applied in the decision moethod of the TS (see Figure 2-4). In this process, if the first objective is not sufficient for resolving difference between two designs, the second objective function values are compared. If the first and second objective function values are equal for two compared designs, the third objective is then used in the decision and so forth. Usually the first objective function is productivity of a design. In such case, a larger throughput is always preferred to a simpler design (lower number of equipment in the design). Similarly, a simpler design is always preferred to a more "top-down" one.

2. 2.2.3 Optimization Algorithm Formulation

A schematic flowchart of the Tabu Search is given in Figure 2-10. In the following, the different rules and options for the algorithm will be discussed.

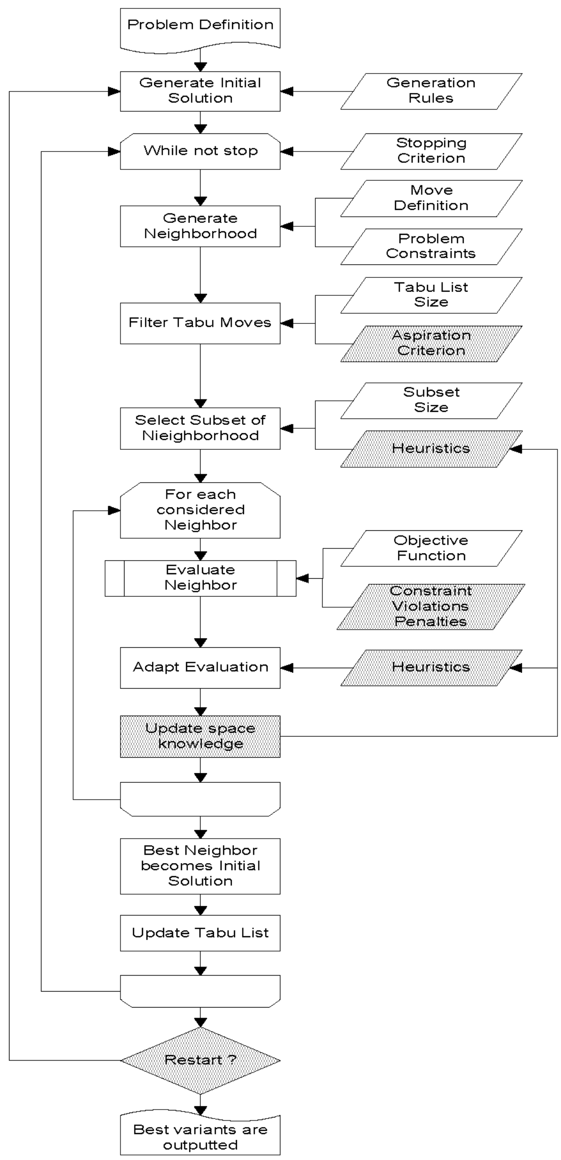


Figure 2-10: Tabu Search algorithm used in the BPD Software (Cavin et al., 2005). Boxes with grey background signify options. Parallelograms represent the rules and the objective functions.

The optimization algorithm provides optimal layouts for the process, while operating parameters are considered fixed.

Equipment units can be assigned to a recipe block in normal mode (one equipment unit operating in the block), parallel mode (two units in parallel are assigned to a block) and serial mode (two and more units operate sequentially within one block). Each recipe block's mode of operation is stored in the design L and is derived from the move applied to the design element L_i. Additionally we differentiate the order of equipment units in sequential mode of operation (in series) and denote units that are at the beginning of the design element $\Uparrow$, end of the design element $\Downarrow$ and within the element $\leftrightarrow$. The double arrows symbol is used for all units assigned to a design element.

For finding the previous or next recipe block in the sequence, the following procedures are defined:

$$k^{+} = find\_next\_block(i, B) \tag{12}$$

$$k^{-} = find\_previous\_block(i, B) \tag{13}$$

Forbidden reuse of equipment in the design layout prevents simple combinatorial arrangement of the problem and introduces additional filter for the free equipment units for block i. In order to identify correctly free units available for each element (L_i) of the design , we define:

- $V_{i,\Uparrow}$ for available equipment units at the beginning of an element, ›
- $V_{i,\leftrightarrow}$ for available units within an element,
- $V_{i,\Downarrow}$ for units at the end of the design element.

Parameter L_{i+1} denotes the next design element (following after L_i), as the last unit of L_i can still be used for processing the block L_{i+1}, if the recipe constraints *[k_i.Constraints]* allow this arrangement. For example, if a transfer is prescribed between the current and previous recipe block, the last unit of previous block cannot be used in the current block L_i.

Three moves are defined for the modification of current design L (Eq. 10):

1. Addition, where one unit is added to L to conduct a specific recipe block (either in serial or in parallel mode)
2. Removal, where a unit is removed from L
3. Replacement as a combination of removal and addition from and to L

Each element in the move list M contains a unit's ID (or two units' IDs in the case of replacement), as well as the block to which it refers and the kind of move it represents.

For a real equipment, an addition move of a unit into a design is subject to the connectivity constraints and hence possible moves are different if the new unit is added in the beginning, within or at the end of a recipe block. For superequipment this restriction does not exist. Free superequipment can always be added into a free slot in the design element, therefore every free superequipment is automatically contained in the list of free equipment V_i for a given element L_i (see Section 2. 2.3 *Superequipment concept*). Addition in series is defined by Eq. 14.

The move is only conductible if the recipe block‘s mode of operation constraints [$k_i.OperationMode$] allow in series arrangement of units within the design element.

$$M_i^{add,s} = \begin{cases} if\ L_i.DesignType = 'parallel' \\ \quad \theta \\ if\ L_i.DesignType = 'single' \\ \quad if\ k_i.OperationMode = 'series\,allowed' \\ \qquad \begin{cases} V_{i,\Uparrow}\,(add\ at\ the\ beginning) \\ V_{i,\leftrightarrow}\,(add\ in\ the\ middle) \\ V_{i,\Downarrow}\,(add\ at\ the\ end) \end{cases} \\ \quad else: \\ \qquad \theta \\ if\ L_i.DesignType = 'series' \\ \qquad \begin{cases} V_{i,\Uparrow}\,(add\ at\ the\ beginning) \\ V_{i,\leftrightarrow}\,(add\ in\ the\ middle) \\ V_{i,\Downarrow}\,(add\ at\ the\ end) \end{cases} \end{cases} \tag{14}$$

Addition in parallel is defined by Eq. 15 and is only possible if the recipe block‘s mode of operation constraints allow for this arrangement.

$$M_i^{add,s} = \begin{cases} if\ L_i.DesignType = 'single' \\ \quad if\ k_i.OperationMode = 'parallel\,allowed' \\ \qquad V_{i,\Uparrow} \cap V_{i,\Downarrow} \\ \quad else: \\ \qquad \theta \\ else: \\ \quad \theta \end{cases} \tag{15}$$

As there are no design blocks possible with *in series* and *in parallel* mode of operation at the same time, the list of all possible addition moves is defined by Eq. 16.

$$M_i^{add} = M_i^{add,\,s} \cup M_i^{add,\,p} \tag{16}$$

Removal of a unit is possible only if the unit is not used both in a previous (k^-) and subsequent (k^+) block, otherwise the removal would produce an invalid design due to the reuse of a unit (see Eq. 17).

$$M_i^{rem} = \left[L_i \mid L_i \notin \left(\left(\bigcup_{j \in k^-} L_{j,\Downarrow} \right) \cap \left(\bigcup_{j \in k^+} L_{j,\Uparrow} \right) \right) \right] \tag{17}$$

It is allowed to remove all units assigned to a block. This increases the flexibility of TS algorithm in low degrees of freedom situations, where only limited number of neighbours is valid for evaluation. Of course, such a design is incomplete and therefore its production rate is zero. Implications for superequipment unit removal are that it is always possible to remove one, if the *no-reuse of equipment* rule is valid (see Section 2. 2.3 *Superequipment concept*).

The replacement moves for a design block L_i are defined by Eq. 18 as a pairwise combination of any removal and any addition. Replacing one superequipment unit with another is preferable, as for example the superequipment unit from previous block can be newly assigned to current block and thus increasing the diversity in the iteration step (see Section 2. 2.3 *Superequipment concept*).

$$M_i^{repl} = M_i^{rem} \oplus M_i^{add} \tag{18}$$

The sum of all possible moves (M) is as follows:

$$M = \bigcup_i (M_i^{add} \cup M_i^{rem} \cup M_i^{repl}) \tag{19}$$

A subset of moves has to be evaluated in each iteration for determining the next neighbour according to TS principles. Eq. 20 shows the subset selection and denotes also that it is forbidden to select moves already stored in the Tabu List (T).

$$M^0 \leftarrow random\ subset\ of\ (M \mid M \notin T) \tag{20}$$

The actual design L is transformed by the application of a move M^m from M° into a new design L^m. The application of all moves in M° to the given design L creates the candidate list W. All designs in W are then evaluated. The evaluation of the design comprises a simulation of the process and the computation of the resulting objective functions.

2. 2.3 Superequipment concept

If considerations about the retrofit problem of additional investment into an existing plant line are to be made, the combinations of equipment on buy list and their characteristics (i.e. unit size, lining material, options, TP range) require exponential solving time. During a standard TS optimization it is necessary to generate a number of combinations in form of designs, where the allocation of each new unit should be varied in the design in order to have a good chance of finding the global optimum.

Superequipment concept has been developed as alternative approach. It simplifies the combinatorial problem, because one superequipment unit substitutes any unit from the list of equipment. Superequipment is not a real equipment. It stands for a model of a unit, where each piece of superequipment can be transformed into a real apparatus in the final design. The method is discussed in (Mosat et al., 2005a).

2. 2.3.1 Superequipment formulation

The superequipment class [S] is defined in the way that any operation from the operation classes present in the assignment matrix A can be conducted in the superequipment:

$$S := \bigcup_i A_i \cdot EqClass \tag{21}$$

Formally we define a *superequipment unit* as an unit belonging to the equipment list E:

$$Superequipment\ unit := (\ E.EqClassID = S\ ) \tag{22}$$

Each piece of superequipment used in a design must be transformed into an existing real unit at some point during the optimization. This transformation is necessary for the evaluation function. That means, the transformation is done in each iteration for each evaluated design and for each superequipment unit. At the same time, the superequipment retains the information about the proposed transformed equipment class for the results postprocessing. These application–specific transformations are explained below.

At the beginning of each iteration and during the design generation, there is no need of knowing the exact equipment class from the equipment classes matrix, nor the size, lining material or TP ranges of a unit. The cycle time, batch size etc. are only needed at the point when objective functions for the specified design are evaluated. If the design L contains any superequipment unit, rules for transformation of a superequipment into a real equipment unit are needed. For each superequipment in design L, an equipment unit type must be available in A. This enables the conduction of all operations from recipe blocks k_i that are assigned to the superequipment:

$$\forall\ (L.Ej.EqClass = S): \bigcap \left(A.EqClass \mid A.OpClass = \bigcup_i L\,.\,E_j\,.\,k_i\,.\,OpClass \right) \neq S \tag{23}$$

The rule excludes conducting incompatible operations in one unit, for example multidrop centrifugation and reaction cannot be conducted in a single unit. On the other hand crystallization, extraction and reaction can all be conducted in a reactor.

If we consider a design L with the composition of n real equipment units E_i and one superequipment unit S_i, the superequipment unit is *transformed* into a real equipment unit according to the transformation algorithm on Figure 2-11, in which the eligible classes are determined according to the recipe block's parameters and other equipment connected to the superequipment. For instance, if a reactor is operating in parallel mode with a superequipment, the superequipment belongs automatically to the reactor class.

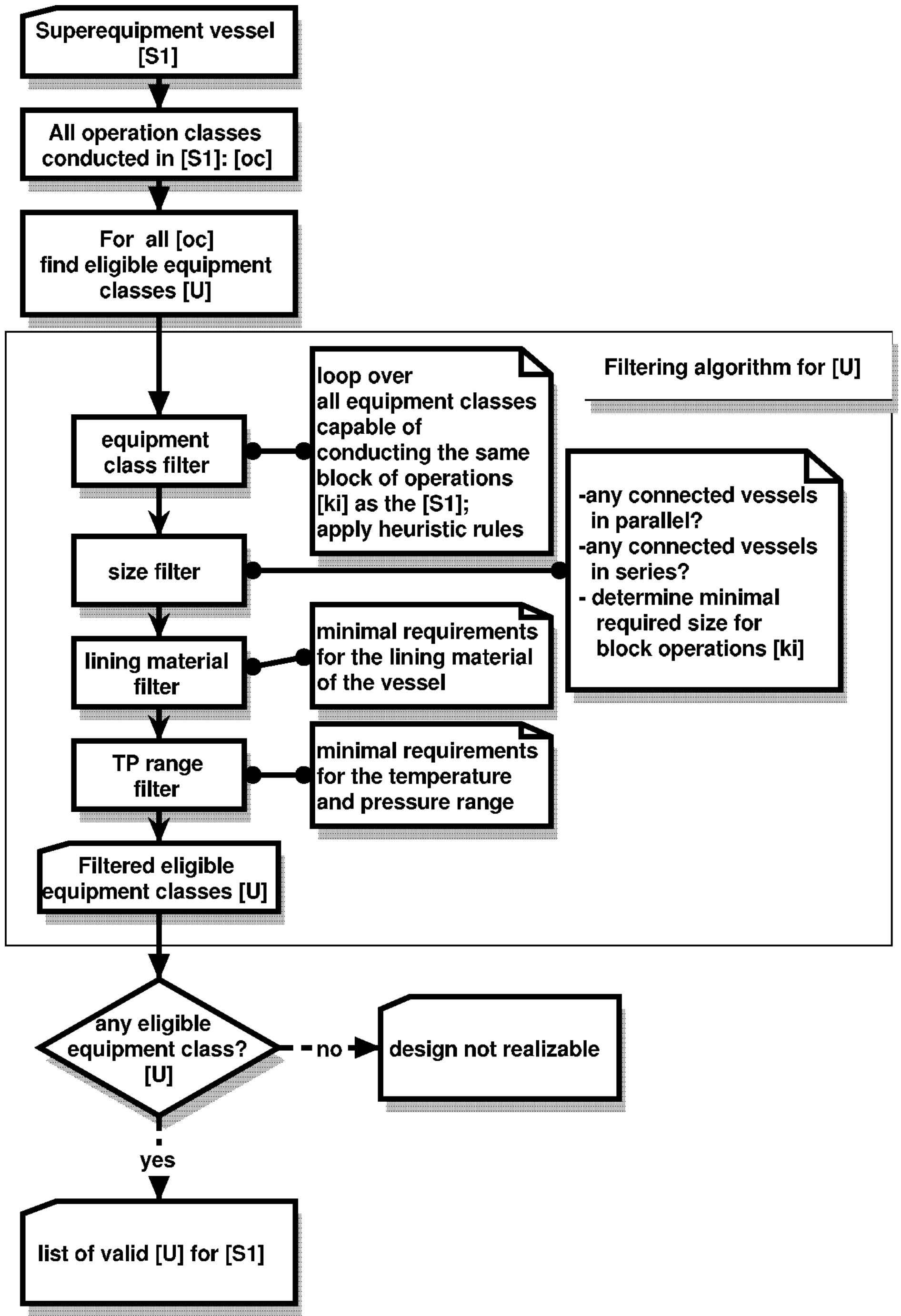


Figure 2-11: Algorithm for transformation of the superequipment unit to standard equipment including filtering according to minimal specification requirements from the recipe block. This transformation occurs prior to objective functions evaluation.

The superequipment transformation is conducted in steps, where all invalid equipment classes are eliminated through partial filters: equipment class filter, size filter, and unit property filter. The first part, limiting the set of equipment classes according to heuristic rules, is related to industrial practice, where parallel operations should be conducted in the same class of equipment with the same characteristics (equal size, lining material, TP ranges) due to quality control. Computation of the size of superequipment unit is related to heuristic and economic criteria. All other properties arise from the minimal requirements of the recipe blocks conducted in the superequipment unit and interconnected equipment. For example: if there is flexibility in the class choice for specific superequipment, selecting the cheaper alternative for the evaluation is preferred. If several equipment sizes are possible (equipment in parallel operations should have the same volume if possible) select the smallest possible size for the unit. Additionally, options for the unit (additional installations, condensers, columns, ...) are determined.

The superequipment transformation example in Figure 2-12 shows three operations, where the second operation is conducted in the superequipment (original design). After transformation we can see one of the possible solutions – a reactor (the middle design). Second valid solution is an extractor (the bottom design). This possible flexibility in the class, size, lining material is maintained through the optimization process up to the final results list, so that the decision maker can see all possible proposals.

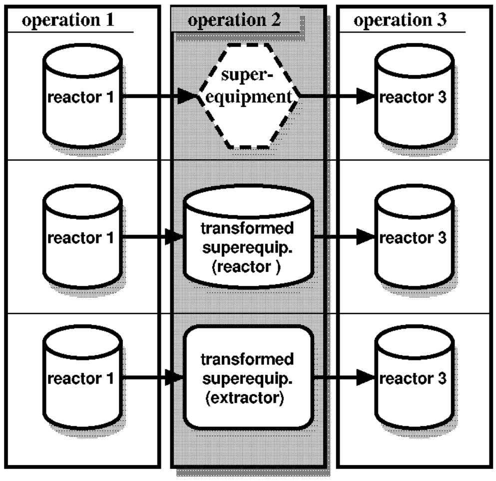


Figure 2-12: Superequipment transformed into a real equipment according to algorithm on Figure 2-11

Figure 2-13 explains the grass–root design process using only superequipment i.e. after fitting the transfers and after obtaining the design consisting only of superequipment units (upper part of the figure). There are two valid possibilities for operation 1 (extraction): extractor or reactor. For evaluation purposes the cheaper unit will be selected for computing the NPV objective function. Second vessel is a reactor of bigger size and third unit has to be a multidrop centrifuge of given type according to the third block specifications. The option of reactor or extractor for the first operation is displayed in the results list.

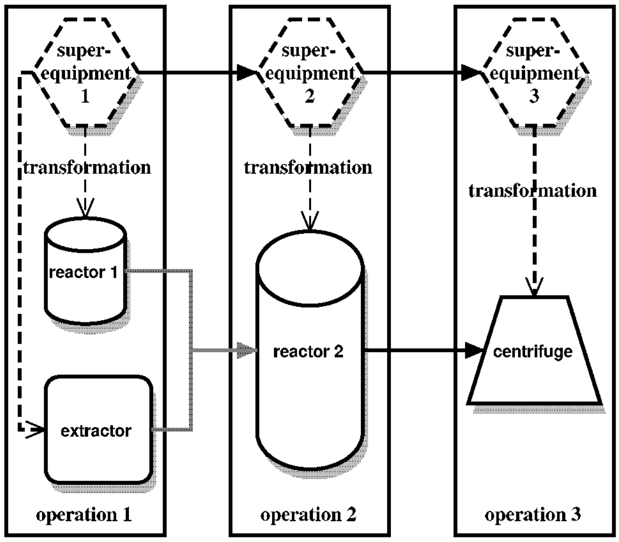


Figure 2-13: Scheme of grass-root design optimization using superequipment concept

The tabu procedures are universally valid for the superequipment concept method and are listed in the section 2. 2.2.3 *Optimization Algorithm Formulation*.

2. 3 Stochastic batch design problem

The problem described in this section is a single product multipurpose batch design optimization, where the single product is to be manufactured in a given plant.

The presented stochastic method poses inclusion of uncertain variables to the multiobjective optimization algorithm input and demonstrates the concept on single product to be manufactured in single multipurpose production plant line under uncertain recipe variables. The uncertain recipe variables can be for instance: operation time and operation volume.

The main aim of this research is to implement a quantitative measure of a batch design response to uncertain operating conditions, which will be referred to as "*Robustness of a design*", or short: robustness.

As a *novel* technique, inclusion of performance robustness as an objective function alongside with productivity of a design results not only in optimal performance design set or solely robust designs, but both optimal performing and robust designs in one.

The deterministic productivity is not satisfactory for making complete decisions on: which from the large number of optional designs to implement in the production stage. The aim of this method is to complement industrial decision making in the early process development stage in order to obtain high productivity and high robustness designs set which, if compared to high-peak productivity - low robust designs, yield stable production quotas under varying conditions.

After discussion with industry experts, we identified many advantages in the *robust design problem* formulation. We decided to examine several robustness criteria as a quantification of performance deviations under uncertain recipe input variables.

2. 3.1 Optimization problem overview

In general, the optimization problem can be expressed in a stochastic two-stage non-linear formulation with recourse (Bastin, 2004).
The *first stage decisions* are those, which have to be taken before the experiment takes place. All these decisions are called the first-stage decisions and the phase in which the decisions are taken is called the first stage.
The *second stage decisions,* also called recourse actions, can be taken after the experiment. They are called second-stage decisions and occur in the corresponding period, second stage.

The aim in this optimization method is to take the first-stage decisions that are in average optimal, with the possibility to take some recourse decisions to face the additional knowledge that will be obtained after disclosure of the uncertainty. This suggests defining an objective function and constraints associated to the first-stage variables, while for the second-stage decisions, an additional objective and constraints that depend on the realization of the random variables are considered. The two stages are then combined by adding the expectation of the second-stage objective to the first-stage objective. The resulting program is called the two-stage stochastic program with recourse.
In the following formulations the first-stage decision variables are denoted as x, second stage variables as y and the uncertain variables as $\xi = \xi(\omega)$, where the probability distribution function is known. The ω is a random event, generally a vector that takes its values from a set of random events Ω and ξ is a real random vector. The optimization problem is defined as follows:

$$\begin{aligned} & min_x \;\; z(x) = f_1(x) + Q(x) \\ & s.\,t. \;\; c_{1,\,i}(x) \leq 0,\, i = 1, \ldots \bar{m}_1 \\ & c_{1,\,i}(x) = 0,\, i = \bar{m}_1 + 1, \ldots m_1 \end{aligned} \tag{24}$$

$$\begin{aligned} & where \quad Q(x) = E[Q(x, \xi)], \quad and \\ & Q(x, \xi) = min_y \; f_2(y(\xi), \xi) \\ & s.\,t. \;\; t_{2,\,i}(x, y) + c_{2,\,i}(y(\xi), \xi) \leq 0,\, i = 1, \ldots \bar{m}_2 \\ & t_{2,\,i}(x, y) + c_{2,\,i}(y(\xi), \xi) = 0,\, i = \bar{m}_2 + 1, \ldots m_2 \end{aligned} \tag{25}$$

The z(x) is the stochastic objective function (in the next Section z(x) is refering to robustness or productivity). The $f_1(x)$ is the first stage deteriministic objective function, $Q(x)$ is the second stage objective function including uncertain parameters and is generally a function of probabilistic error parameters ξ. $E(\xi)$ is the expectation of ξ. The $Q(x)$ is generally measurable. The $f_2(x)$ is the second stage stochastic objective function of uncertain variables y. The deterministic constraints c_1 are used in the first stage and c_2, t_2 stochastic constraints in the second stage. We suppose here that the functions f_2, $t_{2,i}$ *and* $c_{2,i}$ are of the same cardinality. The m_1 symbols denote the fixed constraints, which do not depend on the realization of the random vector. The m_2 constraints of the second stage formulation are associated to the realizations of the random vectors.

2. 3.2 Options, constraints and limitations

The formulation has to fulfil the following constraints and limitations:

- Mode of operation: one equipment unit assignment in normal mode, two equipment units of equal equipment class and equal nominal volumes[5] in parallel mode, two or maximum three equipment units of the same equipment class allowed in series.
- *Continuous utilization of equipment* rule states that according to Good Manufacturing Practice (GMP) it is not allowed during single batch to re-use a once emptied equipment unit. The unit had to be cleaned before re-filling, which leads to increased operation costs.
- Task to equipment assignment: a given chemical task can be performed only in a specific equipment class. For example: reaction can be performed only in reactor class, crystallization either in reactor or crystallizer class.
- Operating conditions: each equipment has one of five lining materials specified and given operation temperature/pressure range. A chemical task can be performed only by equipment unit which meets the resulting requirements. For example: acidic reaction, 5 bar, 180 °C can be performed by PTFE-lined reactor with up to 6 bar and 220 °C. The following lining materials are defined: 1, Stainless Steel V4A; 2, Glass/Enamel/Graphite; 3, Hastelloy; 4, Stainless Steel DIN 1.4539; 5, PTFE
- Material balance and scale-up rules: the material balance is computed in the first-stage of the model, before the optimization. During the optimization scale-up rules are applied if the task is going to be transferred to a unit with different volume. Constant, linear and user-defined scale-up rules were defined for different equipment classes.
- Recipe constraints: if necessary, recipe tasks will be modified to satisfy the quality and safety criteria. The following optional criteria were defined: no transfer between tasks (e.g. if a dangerous compound is present in a unit and transfer would be an additional risk), no in-series mode of operation for current task, no in-parallel mode of operation for current task (e.g. if splitting of the heterogeneous mixture is problematic), specific lining material enforced for current task (e.g. for fluorination reactions).
- Eventually if needed, minimal productivity requirement constraint as a lower

5. The equal nominal volume in parallel mode of operation is optional, but used in praxis. The Good Manufacturing Practice (GMP) regulation requires using two units of the same class in parallel.

bound is used during the optimization, which can be transformed, by help of additional information about the campaign size, into due or delivery date limits. Such constraints are usually defined in connection with economic objective functions, which we discuss in the Section 2. 2.2.2 *Objective functions for multiobjective optimization of batch processes*.

2. 3.3 Multiobjective robust design problem

The common batch design problem focuses on profit and is usually monoobjective, although there are several objectives incorporated within the profit objective function. The primary objective used in this study is productivity defined as amount of product delivered per unit of time [kg/hr] by a single design.

If the profit or productivity of a design are to be maximized, lowering the cycle time and increasing the batch size leads to this objective. Shortening the cycle time of a single batch can be assured by de-bottlenecking the time limiting operations. Increasing the batch size can be achieved by de-bottlenecking the volume limiting operations up to the capacity limit of equipment (scaling-up the recipe, using larger unit), or capacity limit of the plant (using more units in parallel for volume-limiting operations). Problems arise later in the production, after implementing the design, if the design is optimal, but under varying operating conditions the productivity drops. We noticed a systematic behaviour of deterministic global optimal designs: if the productivity of a design is the so-called peak performance, a minor change in the operating variable results in a big drop of productivity. The actual factory productivity difference against the computed value depends on the ratio of design operation time under expected conditions to the operation time under actual conditions. Thus, the design obtained by deterministic optimization will be optimal only and exclusively for the set of parameters. From this point of view, often the designs that rank second or third under deterministic conditions show a lower sensitivity to varying operating conditions.

Therefore we aim at obtaining good performance/productivity/profit designs, but the variations of the performance should be minimal under the specified conditions. Often these uncertain distributions are not known, but using of at least min/max or triangular distributions often reveals hidden bottleneck problems with particular recipe and plant combination if compared to deterministic optimization. Figure 2-14 shows an example of hidden bottleneck in a design, where in the standard deterministic case the operations happening in unit E1 are time limiting, which results in a certain cycle time. After introducing uncertain operation times the operations assigned to unit E3 might become time limiting, and therefore prolonging the cycle time even more.

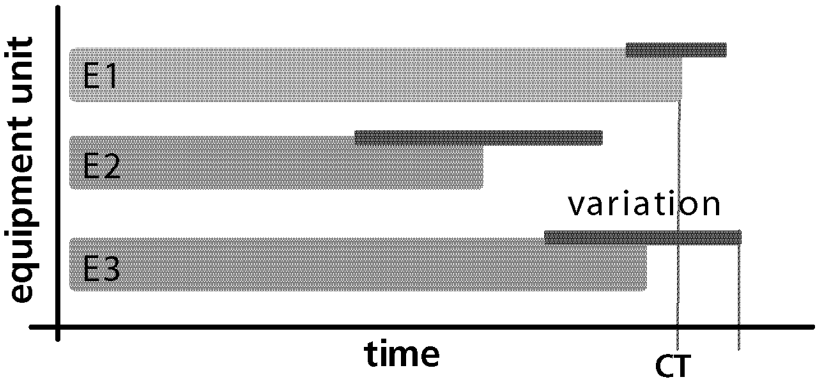


Figure 2-14: Equipment time utilization graph. In the deterministic case, the E1 utilization time determines the cycle time of the design. In the case of time variations (the thin bars above the equipment rectangles) suddenly the cycle time might increase due to a, in a standard case, nonlimiting operation.

The robust batch design problem can be stated as follows: in the assignment of chemical recipe tasks to equipment units find such assignments, which minimize the prioritized set of objective functions. In our formulation, the Equations [24] and [25] are modified for multiobjective optimization for approximated Pareto-ranked domain in a meta-heuristic algorithm. In this case, the prioritized set of objective functions is: 1. productivity of a single design (maximize) 2. robustness of a single design (maximize). The productivity of a design [G] is determined in the first-stage of the algorithm and in general, it is a function of: design variables (L), cycle time and batch size as functions of block recipe variables (B) and uncertain variables (ξ):

$$\begin{aligned} &max \quad G(f(B, \xi), L) \\ &s.\,t. \quad H_{1,\,i}(B, u, L) \leq 0, i = 1, \ldots \bar{m}_1 \\ &H_{1,\,i}(B, u, L) = 0, i = \bar{m}_1 + 1, \ldots m_1 \end{aligned} \tag{26}$$

where $H(u)$ denotes the heuristics defined in the previous section as a function of recipe variables and user-selected variables u.

In this context, a Latin Hypercube method is used for stratified sampling of uncertain states defined in the input recipe matrix B. This sampling approach was selected as a variance reduction technique, in which the selection of sample values is highly controlled, although allowed to vary within the defined interval. The Latin Hypercube stratified sampling method (*Hess et al., 2004, Saliby and Pacheco, 2002*) is based on a full stratification of the sampled probability distribution of each uncertain input variable with a discrete random selection inside each stratum. A random state includes a vector containing one randomly selected value from each sampled variable. The complete resulting random state set is therefore containing the whole range in the definition interval of each uncertain variable. In the case of 3 uncertain variables and 100 stratified intervals, the resulting set will contain 100 random states, in this case 100 different recipe matrices B. In the presented case studies, the productivity is computed for each of the 100 recipes, which results in a discrete function of the productivity density. The robustness measure is then computed according to the discrete productivity probability density function reusulting in a scalar. Different robustness measures investigated in this study are discussed below.

Productivity [kg/hr] as a performance measure is a throughput of the final product based on the selected assignment of recipe tasks to equipment units in a design. If no uncertain variables are specified at the input side, productivity is a scalar value. In the case of relevant uncertain variables in the recipe, such as time of operation or volume of operation, the resulting productivity is computed for each design according to *Latin Hypercube method* from the input variables.
The resulting objective function value is a vector of productivities for different scenarios of the recipe and is denoted by a symbol [G] in contrast to the deterministic scalar productivity value defined in the Section 2. 2.2.2 *Objective functions for multiobjective optimization of batch processes*.

The robustness objective function [*R*] is then determined in the second stage of the optimization algorithm:

$$\begin{aligned} &max \quad R(B, u, L, \xi) = R_{\xi}[G(B, u, L, \xi), L], \\ &s.\ t. \quad H_{2,\,i}(B, u, L) + c_{2,\,i}(G(\xi), \xi) \le 0, i = 1, \ldots \bar{m}_2 \\ &H_{2,\,i}(B, u, L) + c_{2,\,i}(G(\xi), \xi) = 0, i = \bar{m}_2 + 1, \ldots m_2 \end{aligned} \tag{27}$$

where the term denoting the constraints related to uncertain production quota requirements $c_{2,\,i}(G(\xi), \xi) = 0$ can be omitted, as we do not consider such case in this publication.

The number of equipment units used in design objective function can be expressed as:

$$min \quad N_{eq}(L) \tag{28}$$

and this function is not influenced by uncertain parameters in this formulation. The goal is to minimize the number of equipment units used, in order to free a part of the production capacity for eventual future campaigns. The less equipment units used in design, the lower the cleaning and maintenance cost.

2. 3.4 Robustness measures

Robustness of a design [-] is a measure of design productivity stability under varying conditions originating from the recipe uncertainties.

We examine and discuss the following three robustness specifications (compare with Figure 2-15):

1. ***Robustness #1*** (Figure 2-15 (a)) which is expressed as maximal probability value (mode) in the Probability Density Function (PDF) of solution's productivity (Eq. 29). In this case, the mode of the productivity probability distribution (denoted as a function $pdf_G()$) expresses how likely the corresponding productivity will occur among the number of possible probabilistic states given by uncertain

inputs, e.g. time and volume variations of chemical operations. There are two solutions, i.e. designs, displayed in Figure 2-15 (a), where the first solution receives better robustness score. The productivity peak is higher, but the solution cannot achieve the high productivities offered by the second design.

$$R_1[G(B, u, L), L]:$$
$$R_1(B, u, L, \xi) = max[pdf_G(\ G(B, u, L)\ )] \tag{29}$$

2. ***Robustness #2*** (Figure 2-15 (b)) is computed from the most probable productivity point on the *Cumulative Distribution Function* (CDF). If we review the uncertain recipe inputs, we can compute the most probable value ($Y_{i,\ max}$) out of N probabilistic states for each of the input variables (X_i):

$$Y_{i,\ max} = max[\ pdf_G\ (X_i)],\ where\ \ i = 1 \ldots N \tag{30}$$

where $X_i \in \xi$ and *pdf()* denotes probability distribution function of the uncertain variable X_i. If we review the productivity objective function (Eq. 26), we can redefine the generic robustness formulation from Eq. 27 to Robustness #2 as:

$$R_2[G(B, u, L, \xi), L]:$$
$$R_2(B, u, L, \xi) = (1 - cdf_G(G_{max})) \tag{31}$$
$$where\ \ G_{max} = mode(G)$$

where the *mode(x)* function returns an element of a vector x, where the probability density function reaches the maximum. It is used for determining the most probable value of design's productivity G_{max}. The Robustness #2 definition is thus computed from the CDF function of productivity in the point G_{max}.
The Robustness #2 is then defined as: what is the probability, that the given design's productivity is G_{max} or higher?

3. ***Robustness #3*** (Figure 2-15 (c)) takes the most probable productivity of a given solution G_{max} as stated in Eq. 31 and is defined as:

$$R_3[G(B, u, L, Y_i), L]:$$
$$R_3(B, u, L, \xi) = 1 - \frac{G_+ - G_-}{G_{max}}$$
$$where\ \ G_{max} = mode(G) \tag{32}$$
$$cdf_G(G_+) \Leftarrow cdf_G(G_{max}) + c_+$$
$$cdf_G(G_-) \Leftarrow cdf_G(G_{max}) - c_-$$

where the variables c_+ and c_- refer to a predefined bias in the CDF of solution's productivity. Then the Robustness #3 can be expressed as: what is the scaled productivity variation in the interval $<G_-, G_+>$ compared to the expected productivity G_{max} ? The point G_+ is computed from the cumulative density probability in the point G_{max} plus a user-defined bias c_+ (see Eq. 32). The value of G_- is computed in a similar way. This robustness criterion aims at achieving designs with the expected productivity according to the most probable recipe input variables with an addition that if the productivity varies mostly in the predefined cumulative probability interval limit, the design will be awarded a bonus to the robustness. In the case of a *steep CDF* function for a specific solution design around the G_{max} point, the robustness will be high and vice versa.

(a)
(b)
(c)

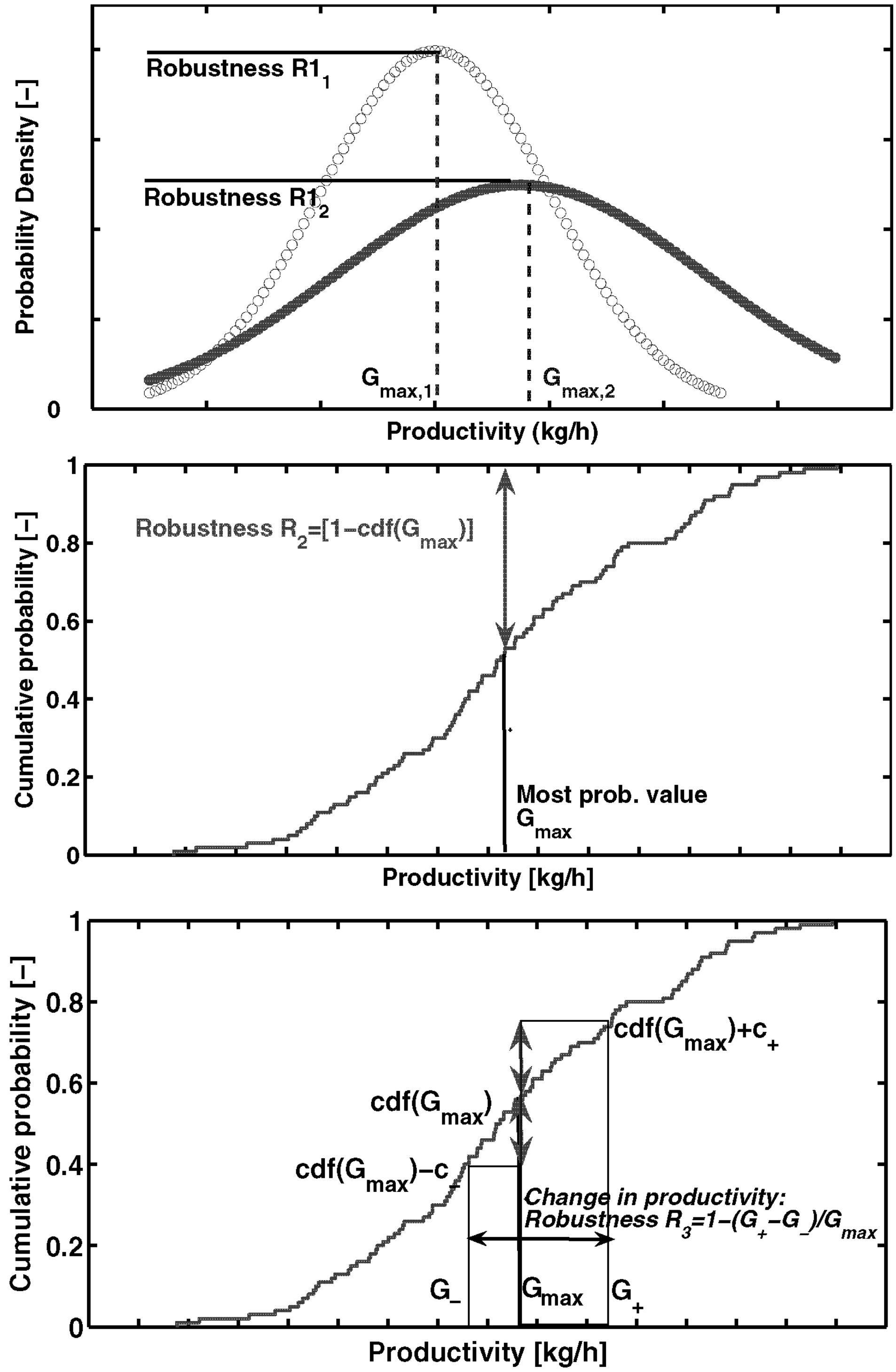


Figure 2-15: Robustness definitions related to a specific design: (a) R1; maximum probability approach, (b) R2; probability of achieving the mode of the productivity or better, (c) R3; robustness as a relative change in the productivity computed from the productivity mode probability value, and a specified cumulative probability distribution range ($cdf(G_{max})$ - c_- , $cdf(G_{max})$ + c_+).

2. 3.5 Handling uncertainty within Tabu Search optimization

This section lists the TS options used specifically in the Chapter 4 *Uncertainty with application to design robustness measures.*

Probability distributions Our method allows for using any custom probability density function, inclusive measured point-sets. For the purposes of the method demonstration, the two most common probability density functions (PDF) will be used:

- Triangular PDF with notation:

$$tripdf(A, B, C) \tag{33}$$

  where A < B < C and A, C are parameters with a scaled relative probability of 0, B is a parameter with a scaled relative probability of 1. Accordingly, the function *trirnd(A, B, C)* will choose random numbers from the triangular distribution.

- Normal probability density function referred as:

$$normpdf(\mu, \sigma) \tag{34}$$

  returns the PDF of the normal distribution with mean μ and standard deviation σ. The random function *normrnd*(μ, σ) returns an array of random numbers chosen from a normal distribution with mean μ and standard deviation σ.

- Lognormal probability density function referred as

$$lognpdf(log(\mu), \sigma) \tag{35}$$

  returns the PDF of the normal distribution with mean μ and standard deviation σ. The random function *lognpdf*(μ, σ) returns an array of random numbers chosen from a lognormal distribution with mean μ and standard deviation σ.

The neighborhood The neighbourhood W of a solution L (a design) is a set of all solutions created by applying all defined moves M:

$$W(L) = L \leftarrow M \tag{36}$$

A random sub-neighbourhood is selected from the current solution candidate neighbourhood in each iteration. The size of the sub-neighbourhood is dependent on: total number of neighbours, total number of non-tabu and tabu listed moves. A reasonably large fraction of moves is selected for evaluation dependent on the size of the problem and CPU power.

Types of moves The moves applied to a design are chosen randomly and with special consideration of the tabu tenure. In each iteration of the TS algorithm, the given design solution is modified by applying a number of selected moves that represent the modification of one recipe task-to-equipment assignment. The following moves are provided:

1. change operation mode to "*normal*"
   if operation mode is not *normal*, eliminate all equipment units except of one and change the mode of operation
2. change operation mode to "*in parallel of two equipment units*"
   if the operation mode is not *in parallel*, add or remove equipment units to a total count of two and change the operation mode to parallel.
3. change operation mode to "*in series of two equipment units*"
   if the operation mode is not *in series of two equipment*, add or remove equipment units to a total count of two and change the operation mode to *in series of two equipment*.
4. change operation mode to "*in series of three equipment units*"
   if the operation mode is not *in series of three equipment*, add equipment units to a total count of 3 and change the operation mode to *in series of three equipment*.
5. *remove one equipment unit* (if possible)
   remove one of the equipment units from the current operation. In the case of *in series of three equipment*, the operation mode becomes either *in series of two equipment* or *in parallel*, with the preference of the latter, depending on the condition that the two equipment units in parallel must be of equal class and equal size.
6. *add one equipment unit* (if possible)
   adds one equipment unit to an operation task
7. *exchange one equipment unit* (if any feasible unit is free for such move)
   remove randomly one assigned unit from a recipe operation and add another randomly selected unit from the free equipment group, which has to be of suitable-equipment class. It has to be noted, that the term free equipment group is specific to each recipe operation in the design. One equipment can be assigned to perform many operations, as long as the continuity in usage is maintained.

Tabu Search options Selected Tabu Search options are applied for improving the convergence rate of the algorithm. The aspiration criterion, selective steepest ascend (i.e. choose always worst solution for a specified number of iterations) and random restarts were used.

Sorting criteria Comparison of two or more designs according to multiple objective functions is performed via the importance function: 1. global optimum, 2. dominance (approximated Pareto-optimum), 3. local optimum in one of the objective functions, 4. good-performing solutions according to user specifications. The problem has to fulfil the constraints defined in the Section 2. 3.2 *Options, constraints and limitations*.

Superequipment application

3

Abstract

A novel approach for solving different design problems related to single products in multipurpose batch plants is presented: the selection of one production line out of several available, additional investment into an existing line or plant, and grass-root design of a new plant. Multiple objectives are considered in these design problems. Dominating (approximated Pareto-optimal) solutions are generated by means of a Tabu Search algorithm. In the novel approach the concept of superequipment has been defined as an abstract model, which is capable of performing any physico-chemical batch operation. Each superequipment is transformed into a real equipment unit, for example a reactor, during or after the optimization in order to evaluate performance parameters of a design. This novel concept uses an implicit definition of a superstructure and essentially optimizes on the transfers between different equipment units in a design.

On the basis of case studies we demonstrate that the application of the superequipment concept offers a number of advantages for the investigated design problems. For example, in the evaluation of investment into single equipment pieces to be added to existing plants or production lines only the maximum number of additional equipment, each represented as a superequipment, has to be specified instead of a list consisting of a higher number of explicit units. Similar advantages arise for grassroot design problems or for the selection of a production line or plant out of several that are available for the production of a specified chemical.

The comparison with optimization results obtained with a conventional Tabu Search algorithm revealed that the superequipment approach is capable of identifying the dominating approximated Pareto-optimal solutions in significantly reduced computation time.

3. 1 Introduction

The superequipment concept, stipulated in the Section 2. 2.3 *Superequipment concept* introduces a mathematical model of a unit capable of performing a role of any equipment unit combination. In this chapter three case studies studying this concept will be presented. An introduction to the problem has also been presented in (Mosat et al., 2005b).

The first case study offers insights into retrofitting a small monoproduct batch plant by buying at maximum two new equipment units with detailed specifications and predictions about the production increase and the resulting NPV after the investment.

The second case study studies the multiple plant lines selection out of several available for a pharmaceutical product, which should be manufactured by a design with high productivity and four other important objectives. A comparison of standard TS optimization invoking the same optimization procedure several times versus a single superequipment optimization is performed showing clearly advantages of the latter method.

The third case study demonstrates using a large number of superequipment units. A grass-root design of a batch plant is examined for a given recipe.

3. 2 Case studies

3. 2.1 Investment scenario for an existing plant and a given recipe – L- Ascorbic Acid case study

Figure 3-1 displays a principle of a conventional optimization method, where for each additional equipment, which will be considered as an investment option, a new optimization has to be performed. In case of three additional considered units, such procedure requires manually defining three optimization problems, running three times the TS optimizations as well as reviewing and filtering three different results lists. For instance three equipment units could be defined as: 1. reactor, 6 m^3 nominal volume, PTFE lining material, up to 6 bar pressure, without mounted condenser, 2. crystallizer, 10 m^3 nominal volume, stainless steel lining material, up to 2 bar pressure, 3. multidrop centrifuge, 400 l nominal volume, Hastelloy lining material, 2 bar pressure. As we see, considering all combinations of equipment classes, nominal volume, lining and construction materials, additional options and more is not feasible and a new method for handling such problem types needs to be used. That means for instance all reactor sizes combined with all construction and all lining materials with condenser attached or without,

in total only for reactors there are 216 of such combinations.
On the right-hand side of Figure 3-1, the superequipment concept for this problem setting requires only setting-up one optimization problem, where one superequipment unit representing large number of real existing units has to be defined. In such procedure not only three, but all good-performing equipment units will be considered in the TS optimization automatically. At the end, one results list is obtained and it contains a sorted and filtered list of the most beneficial equipment unit combinations.

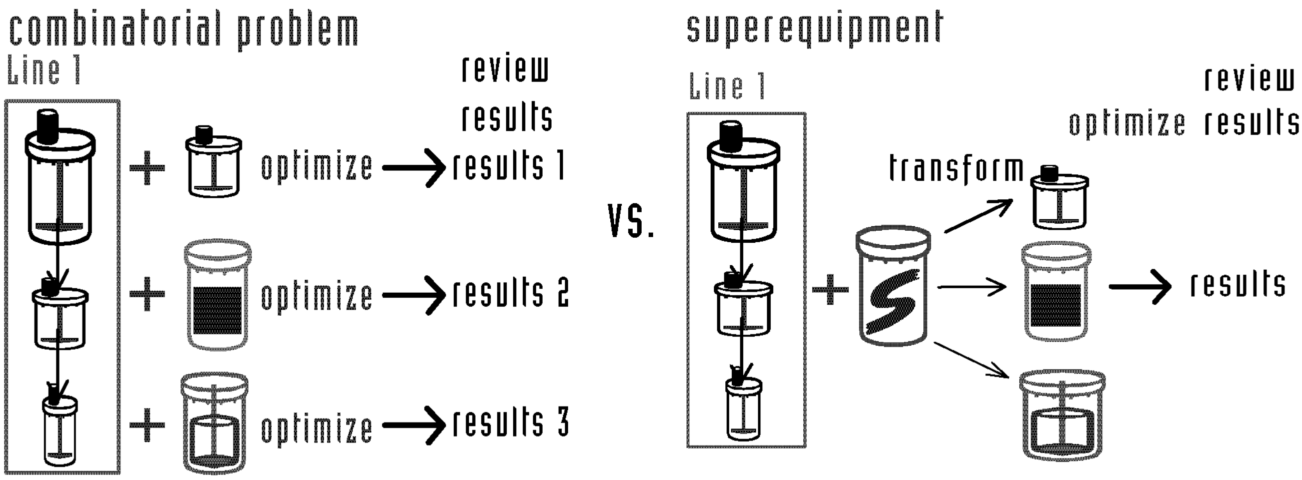


Figure 3-1: Investment into an existing plant; problem methodology demonstration. Comparison of two methods, on the left the conventional TS combinatorial optimization problem, on the right the superequipment optimization problem.

The Vitamin C case study is based on the (Reichstein and Grüssner, 1935, Oster and Fechtel, 2002) synthesis and was selected for its simplicity to demonstrate the basic principle of superequipment concept. Similar case study has been published by (Niedrig, 2004) where the estimation of equipment costing is studied in more detail and optimized for three different scenarios.

We define a small batch plant (Plant C4) and examine investment possibilities in order to deliver the product and increase the plant production capacity for the future campaigns. The aim is to produce and sell 1000 t of the product within 5 years time constraint. As for this case an investment is necessary, additional objective is to maximize the NPV of a project with equipment investment considering additional constraints discussed below. There is only space for two additional unit installations within the production building. The productivity was selected as primary and NPV as a secondary objective function, because of a large penalty if the delivery time frame of 5 years cannot be fulfilled. A comparison of results obtained by superequipment concept and conventional TS optimization is presented.

3. 2.1.1 Recipe description

According to (Reichstein and Grüssner, 1935), vitamin C is produced from D-glucose. Since the transformation step from D-Glucose to L-Sorbose includes a microbiological oxidation reaction, the first step of the Reichstein synthesis was left out and it is assumed that the L–Sorbose can be bought as a raw material and the production proceeds from this reactant. Therefore the presented case study included only the remaining steps (see Figure 3-2) of the original Reichstein route.

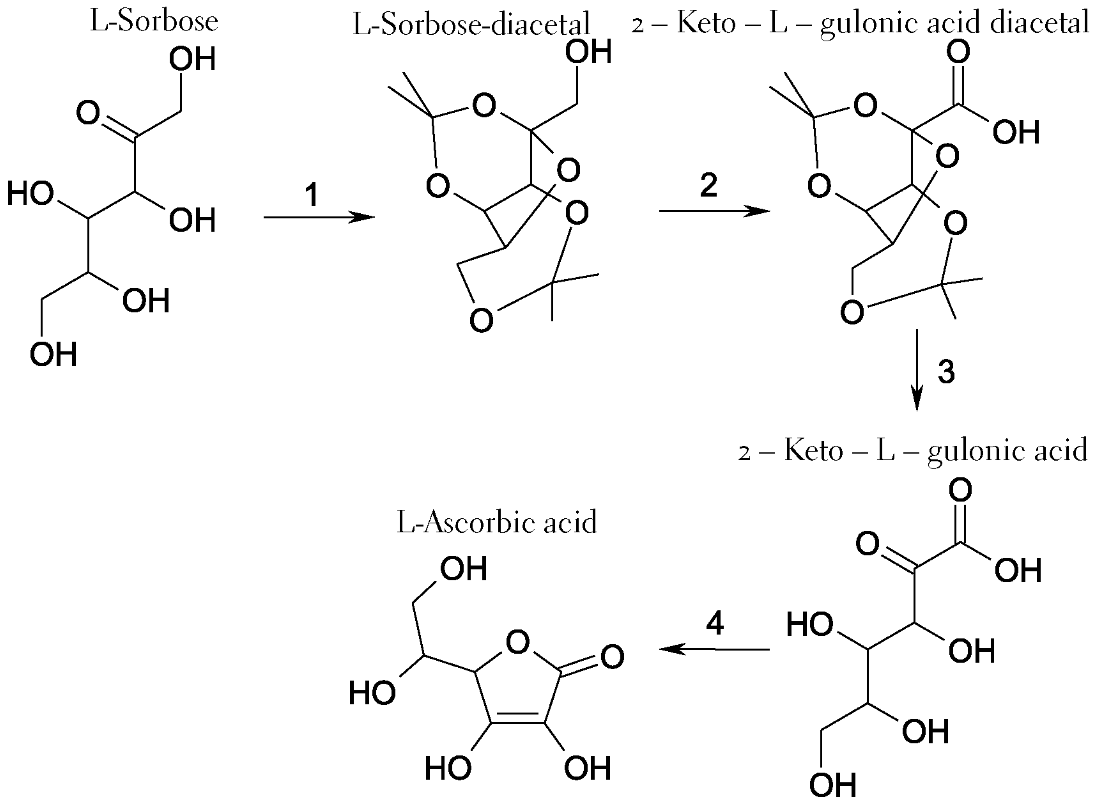


Figure 3-2: Vitamin C case study reaction scheme.

A rough outline of the process and the utilized chemical reactions are shown below, for the complete Batch Plus™ recipe see Appendix A-3 *Vitamin C Reichstein synthesis in Batch Plus™ software.*

Step 1: L–Sorbose to L–Sorbose–diacetal

This step consists mainly of the introduction of the diacetal protection group for the later oxidation to carboxylic acid (see Figure 3-3). A reactor is charged with L–Sorbose and acetone. Then some sulfuric acid is added as catalyst and dehydrating agent. During the reaction, the mixture is kept at 4°C. After the reaction, it is neutralized with sodium hydroxide. The excess of acetone is distilled and recycled whereas the diacetal is extracted with toluene (block 1).

$$C_6H_{12}O_6 + 2\ C_3H_6O \xrightarrow[4\ °C]{H_2SO_4,\ conc.} C_{12}H_{20}O_6 + 2\ H_2O$$

Figure 3-3: Transformation of L-Sorbose to L-Sorbose-diacetal

Step 2: L–Sorbose–diacetal to 2–Keto–L–gulonic acid diacetal

The contents of the reactor from Step 1 are transferred into another one. Then a catalytic amount of nickel sulfate is added along with some sodium hypochlorite as an oxidizing agent. This oxidation reaction (see Figure 3-4) is performed at 60 °C and the resulting 2–Keto–L–gulonic acid diacetal is isolated by acidification and extraction with sulfuric acid (block 2).

$$C_{12}H_{20}O_6 + NaOCl \xrightarrow[60\ °C]{NiSO_4} C_{12}H_{18}O_7 + NaCl + H_2$$

Figure 3-4: Transformation of L-Sorbose-diacetal to 2-Keto-L-gulonic acid diacetal

Step 3: 2–Keto–L–gulonic acid diacetal to 2–Keto–L–gulonic acid

This step contains the removal of the diacetal protection group, which is done by hydrolysis (see Figure 3-5) at 90 °C (block 3). The intermediate is separated from the released acetone by crystallization (block 4).

$$C_{12}H_{18}O_7 \xrightarrow[90\ °C]{2\ H_2O} 2\ C_3H_6O + C_6H_{10}O_7$$

Figure 3-5: Transformation of 2-Keto-L-gulonic acid diacetal to 2-Keto-L-gulonic acid

Step 4: 2–Keto–L–gulonic acid to L–Ascorbic acid

The intermediate from the previous step is dissolved in toluene and cyclization to L–Ascorbic acid in the presence of hydrochloric acid (as a catalyst) takes place (block 5) (see Figure 3-6). The resulting Vitamin C is then dissolved in water, crystallized (block 6) and centrifuged in a multidrop centrifuge (block 7). Recrystallization of final product takes place as a purifying step in (block 8). It is assumed, that the drying step processes product crystals from multiple batches in a distinct drying line, therefore it will not be included in the case study.

$$C_6H_{10}O_7 \xrightarrow[30\ °C]{HCl,\ conc.} C_6H_8O_6 + H_2O$$

Figure 3-6: Transformation of 2-Keto-L-gulonic acid to L-Ascorbic acid (vitamin C)

3. 2.1.2 Plant lines description

Plant C4 *(base plant)*

Reactors: 5; 6.3 m^3: 4; 10 m^3: 1; one 6.3 m^3 reactor and one 10 m^3 reactor have a distillation column attached

Multidrop centrifuge: 1; 1.2 m^3, vertical basket

Various additional units: tanks, filters

Plant C4S *(base plant and superequipment)*

Two additional superequipment pieces have been added to the base plant C4 in order to simulate investment options.

Plants *C4r, C4ce, C4rce (base plant plus additional units)*

C4r consists of the base case plant C4 plus an additional 10 m^3reactor

C4ce consists of the base case plant C4 plus one multidrop centrifuge 1.2 m^3

C4rce consists of the base case plant C4 plus one additional 10 m^3 reactor and one multidrop centrifuge 1.2 m^3

3. 2.1.3 Optimization settings

By optimization of plant C4 with additional equipment units, we increase productivity as compared to the base case design. The limit of maximum two new units should represent the space limitation in the multipurpose production plant. We demonstrate the usage of the E_{buy} list, which in this case consists of: max. 2 reactors, max. 2 multidrop centrifuges, which can be bought and installed into the plant C4. The constraint of at most two significant units as an investment is also part of the E_{buy} list (condensers count as an option for reactors). The E_{buy} list represents a set of additional constraints for the problem. In this case it is used at the beginning of the optimization (adding 2 superequipment units into a base plant) and in the transformation algorithm of superequipment into a real unit, where only the listed options are allowed for transformation.

The prioritized list of objective functions for this case has been set as follows:

1. productivity, 2. NPV, 3. special diversification function, 4. batch size, 5. number of equipment units. (see Section 2. 2.2.2 *Objective functions for multiobjective optimization of batch processes*)

The objective functions for the base case (plant C4) have been set as:

1. productivity, 2. campaign costs, 3. batch size, 4. number of equipment units, 5. floors–up indicator.

From previous examinations the optimization settings for Tabu Search have been set as follows: neighbourhood size 30, the tabu list length is 70, forbidding shortly visited neighbourhoods. Stopping criteria for the optimization were set to 60 iterations after finding last non-dominated solution and there were 28 restarts during the run, as no improvement has been achieved in the last 10 restarts.

Economic data

The data for determining costs of the campaign and NPV have been obtained from various sources (Schnell_Publishing, 2004), mostly from industry experts. The production plan is to produce 1000 t of Vitamin C under 5 years delivery time. Projected selling price is 10.0 USD/kg of Vitamin C and remains constant over time. Reactants, solvents and other chemicals costs are 4.73 USD per kg of product. We assume 8000 hr/year operation time. Total costs of campaign are to be determined for each design separately and include price of utilities, overhead, changeover, waste management, rent for plant per hour, labour and other. All of the cost categories are customizable and reflect current market prices. Campaign costs do not include potential investment. In the NPV computation, the cash flows include the complete eventual investment in equipment units, although we are considering a multipurpose batch plant with varying product portfolio and also later products benefit from this investment. The investment loan begins 6 months before the campaign beginning.

3. 2.1.4 Results and discussion

Results for base case plant C4 – conventional optimization

Optimization results show, that with current base plant (Plant C4) we can achieve production rates up to 20.2 kg/hr by implementing a design with 6 equipment units. Optimal production costs in the base plant C4 are 7.98 USD/kg of product including material costs. The simplest possible design employs 4 units and can manage all eight blocks of the recipe.

For the following considerations, we assume that the current production of Vitamin C is performed by design ranked #1 in the list of results (containing 150 dominating and structurally diverse designs), which has a batch size of 240 kg/batch and a productivity of 20.2 kg/hr. The corresponding project duration is 6.2 years, which is too high and suggests an additional investment.

Review of this design (see Fig. 3-7) shows that the recipe block 1 is performed in parallel (reactors R6d1, R6d3), block 2 in normal mode of operation (reactor R6d5), blocks 3, 4, 5 and 6 in reactor R10d4, centrifugation from block 7 in multi-drop centrifuge Ce1.2d1 and finally the crystallization from block 8 in reactor R6d2.

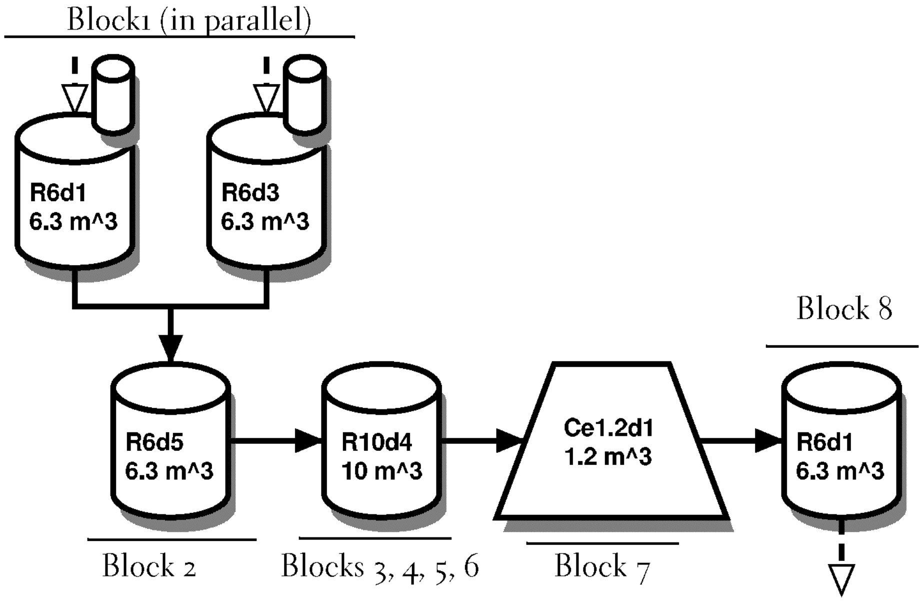


Figure 3-7: Design #1 from plant C4 (base case) in Vitamin C case study as obtained by conventional TS optimization; R denotes reactor, Ce centrifuge. The volume limiting unit is R10d4 filled with 10 m^3 of reaction mixture, and the time limiting unit is Ce1.2d1 with 730 min utilization time.

Superequipment Optimization of Investment Scenario – Results for Plant C4S

The following step in examining possible investment options includes adding two superequipment units into the existing plant C4, which means that the algorithm will generate either designs utilizing zero, one or two pieces of superequipment in addition to the units from plant C4. If the resulting optimal design contains no superequipment, an additional equipment on top of plant C4 would not improve the profitability of the process.

Table 3-1: Subset of dominating (assumed Pareto–optimal) designs for plant C4S in Vitamin C case study by TS optimization using superequipment method

Design rank [#]	Investment	Equip. size [m^3]	Investment [kUSD]	Productivity [kg/hr]	Nr. of units [pcs.]	Payback time [year]	NPV [kUSD]	Campaign time [years]
1	react+centr.	10; 1.2	820	27.8	8	6.6	1300	4.1
35	centrifuge	1.2	340	23.1	7	12.5	1130	4.9
49	react.+cryst.	16; 6.3	800	21.3	8	7.2	1180	5.4
69	no investment	--	0	20.2	6	-	1520	6.2

However, the algorithm with superequipment method identified several profitable possibilities for investment. A selection of investment options is listed in Table 3-1, where the designs are dominating (assumed Pareto–optimal) in the listed objective functions and have been chosen from a results list with 200 entries. If we compare the Design #1 and Design #69, the NPV is the highest for the no investment case. However the project time constraint 5 years must be taken into account. Thus from the listed solutions only the Designs #1 and #35 are satisfying this constraint.

The best performing design according to productivity utilizes additionally one 10 m^3 reactor and one 1.2 m^3 centrifuge (see Fig. 3-8). Reactor specifications determined by the operation requirements are: steel construction material, stainless steel lining/V4A, temperature range −15 °C to 150 °C, up to 6 *bar* pressure resistance and no distillation column. The centrifuge is sold with PTFE lining and a buffer tank.

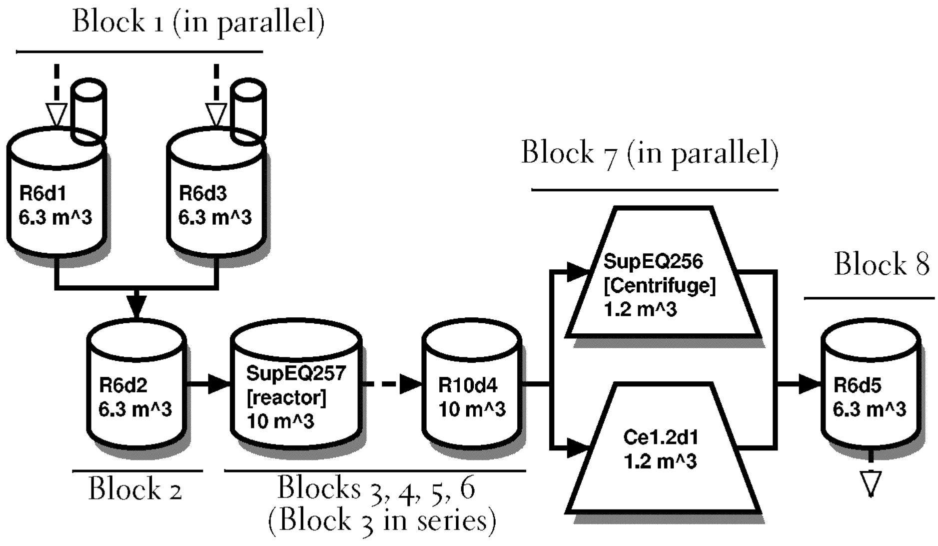


Figure 3-8: Vitamin C case study with investment as obtained by superequipment method optimization. Design ranked #1 from plant C4S. Dashed arrow refers to *in series* operation mode.

Objective function values are: productivity 27.8 kg/hr, batch size 0.24 t/batch, and payback time of 6.6 years. Time analysis (Figure 3-9a) shows shortening of cycle time from 730 min to 525 minutes by better utilizing the equipment and parallel assignment of the two centrifuges. The volume utilizing analysis (Figure 3-9b) shows the volume limiting equipment (R10d4 and SupEq257), which perform the operation in series. The centrifuges are never volume limiting, therefore the volume utilization is not shown.

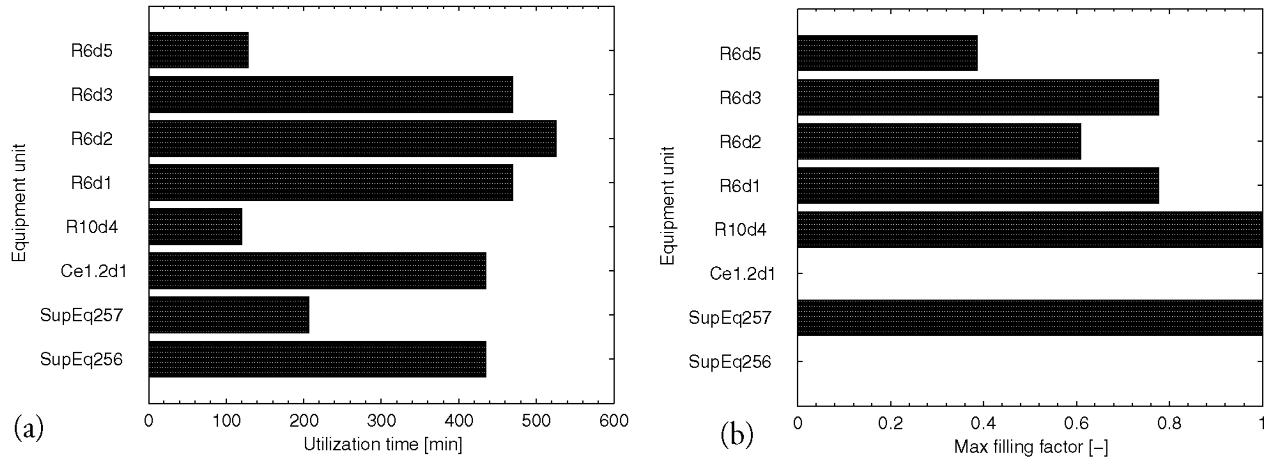


Figure 3-9: Vitamin C case study with investment as obtained by superequipment method optimization. Design #1 in plant C4S, (a) equipment time utilization, (b) equipment volume utilization.

Comparison of Conventional TS Optimization (Plants C4r, C4ce, C4rce) and Superequipment Method Optimization (plant C4S)

For comparison we studied investment options using a conventional TS optimization approach (Cavin et al., 2004, Cavin et al., 2005) and accordingly defined three new plant lines mirroring the investments on top of the base plant: plants C4r (additional reactor), C4ce (additional multidrop centrifuge) and C4rce (reactor plus multidrop centrifuge), see Table 3-2.

Table 3-2: Investment scenarios for plant C4; Vitamin C case study and conventional TS optimization

Plant name	Equip. type	Equip. size [m^3]	Investment [kUSD]	Options
C4r	reactor	10	480	distill. column
C4ce	multidrop centrifuge	1.2	340	buffer tank
C4rce	reactor+centrifuge	10; 1.2	820	dist. column + buff. tank

The optimization of each case separately provides in total three distinct result lists with 100 designs for each case. The best designs according to productivity have been selected for comparison (see Table 3-3).

The new 10 m^3 reactor in the plant C4r provides only a small improvement over the base case. All equipment is used in this plant and a productivity of 20.7 kg/hr is only about 1 % higher than in the base case.

Table 3-3: Comparison of optimal designs from plants C4, C4r, C4ce, C4rce, Vitamin C case study; conventional TS optimization

Plant name	Prod. rate [kg/hr]	Floors up [-]	Batch size [t]	Nr. equip. [pcs.]	Campaign costs [mio. USD]	NPV [kUSD]	Campaign time [years]
C4	20.2	4.5	0.24	6	7.98	1520	6.2
C4r	20.7	6.0	0.24	7	7.90	1020	6.0
C4ce	23.1	4.5	0.24	7	7.86	1130	4.9
C4rce	27.8	4.5	0.24	8	7.79	1300	4.1

A 1.2 m^3 centrifuge in the plant C4ce offers 12 % capacity improvement over the base case, which has been achieved by ordering two identical centrifuges in parallel and thus shortening the original cycle time from 730 minutes in the optimal design from plant C4 to 525 minutes in the design from plant C4ce, where the time limiting operations take place in one 6.3 m^3 reactor (single mode of operation).

The combination of centrifuge and reactor in the plant C4rce offers massive shortening of the cycle time by employing second centrifuge in parallel, increasing the utilization of equipment and decreasing the cycle time even more by serial arrangement of the units R6f2–R10e1 and R10e1–R10c1 (see Figure 3-10).

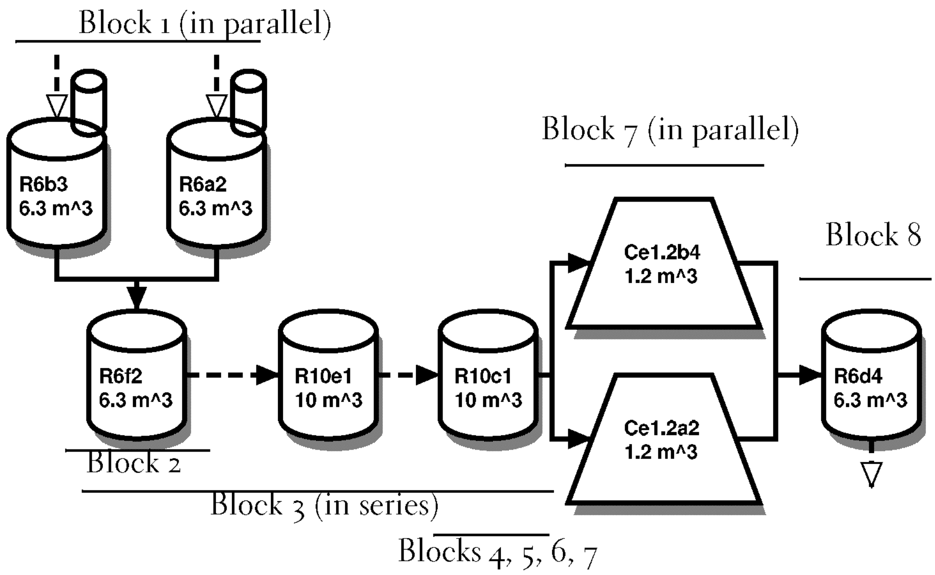


Figure 3-10: Vitamin C case study with investment as obtained by conventional TS optimization. Design #1; plant C4rce. Dashed arrows refer to in series operation mode.

Comparing the optimal design according to productivity from plant *C4rce* (Figure 3-10) and the superequipment method optimized design on Figure 3-8, identical arrangement of the equipment in both flowsheets can be observed. Although the flowsheets are similar, there can be differences in the recipe block assignments. For example in the design #1 from plant *C4S* (Fig. 3-8) the block 2 is performed in R6d2, block 3 (in series) in SupEq257 and R10d4 and block 4 in R10d4 . In the flowchart of design #1 from plant *C4rce* (Fig. 3-10) the block 2 is performed in R6f2, block 3 in series in R6f2, R10e1, R10c1 and the block 4 in R10c1. Both layouts have the same objective function values.

All designs from Table 3-3 were identified in the results list from the superequipment method optimization and prove the superequipment method reliable.

3. 2.1.5 Conclusions

The non-dominated solution[6] is found in both conventional TS approach and by using superequipment concept with TS. Comparison of the data from conventional optimization and superequipment method shows that the productivity in both cases is the same for the same designs.

The flowcharts of the corresponding optimal designs in both methods are identical in location of volume limiting equipment, time limiting equipment and structure of designs with regard to the same type of equipment. The differences in block allocations in series have no effect on objective function values.

Comparison of the CPU time needed for performing three separate optimizations against one optimization utilizing the base plant and two pieces of superequipment shows clear advantage of the latter method: for each standard optimization the mean value of duration for 1000 iterations was 856 s, the mean value of time required for finding the non-dominated optimum was 3510 s. The superequipment method requires more computational time for validation and transformation algorithms and the mean value of duration for 1000 iterations is 2051 s. The mean value to find the non-dominated optimum is 2430 s, because the problem utilizing superequipment units is less constrained and requires less iterations to reach a global optimum. Furthermore the superequipment approach has the significant advantage that only the maximum number of additional units has to be specified for the investment scenario while an explicit definition of a larger number of equipment is required in the conventional approach. This reduces the amount of work related to input specifications, results list compilation, and number of optimization runs.

3. 2.2 Plant selection for given recipe — 4-(2-quinolinylmethoxy)-phenol (Product H) case study

The aim of this case study is to show how in a single run diverse plant lines can be optimized by superequipment method and compared with each other. As a proof of functionality, we optimize the three plant lines by standard method and compare with results obtained by superequipment optimization.

6. Numerous optimization runs from different random starting conditions have been performed and no better solution could be found within the given time-frame (maximum solving test time was 48 hours), which suggests that the solution is, with a high probability, a global optimum.

Figure 3-11 shows schematics of such optimization. In the standard case one would select a number of existing batch plant lines from corresponding databases, optimize multiple times and review multiple results lists in order to obtain the best design which will be performed in one of the plant lines. On the other hand, superequipment methodology for batch plant line selection requires: superequipment domain specification (saying how many superequipment units will be needed for current optimization) plus the databases of the considered batch plant lines. Afterwards only one optimization run will be performed delivering sorted results list consisting of top-performing, in general dominating (approximated Pareto-optimal), designs for all considered plant lines. Thus the number of optimization runs is reduced to one.

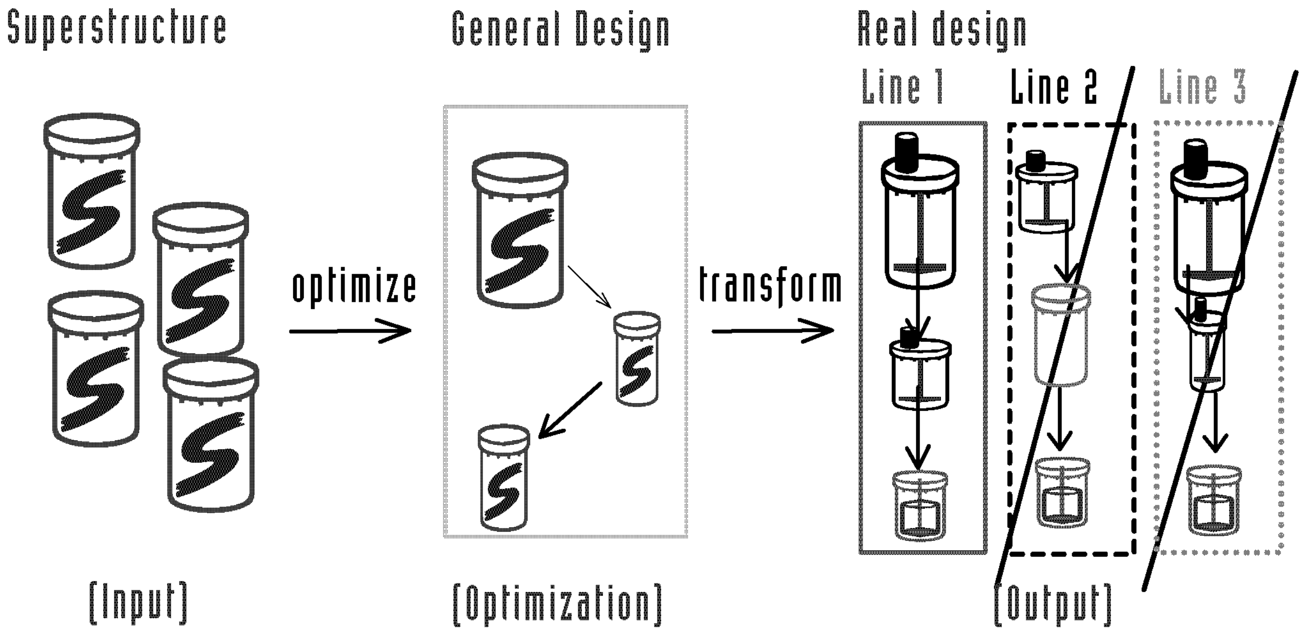


Figure 3-11: Plant line selection procedure using predefined superstructure consisting only of superequipment units plus predefined multiple batch plant lines containing real equipment units delivers results for three batch plant lines in one optimization run.

This case study shows production of a fine chemical used in the pharmaceutical and photo industry. The synthesis is based on a reactant known under commercial name Quinaldine which is freely available on the market. The recipe and basic process simulation of 4–(2–quinolinylmethoxy)–phenol (or its sodium salt), referred to as *Product H* have been presented by (Petrides et al., 2002). An overview of the chemical process can be found in Appendix A-4.

3. 2.2.1 Recipe description

The recipe consists of 33 individual steps ordered into 12 blocks (this is equivalent to the recipe matrix *B*). The Quinaldine as a key component is transformed through 5 reaction steps into product H. None of the reactions needs extreme temperatures or pressures and can be easily controlled. Minimal number of units for this recipe is 9. The reaction scheme listing the most important reactions of this procedure and the BatchPlus recipe is listed in the Appendix A-4 *Quinaldine derivate synthesis - Product H*.

Charging the reactants, evacuating the reactor and first reaction are assigned to (block 1). Quality control test is performed at the end of reaction.

Second reaction step is performed in the (block 2).

Extraction takes place (block 3) and third reaction step follows after charging with additional reactants (block 4). Also in this reaction, quality control is needed.

Crystals of the intermediate product have to be filtered and washed (block 5).

After dissolving the intermediate with additional solvent, reaction 4 takes place (block 6). Another filtration of the crystals (block 7) is followed by dissolution and reaction 5, where the final product in solid state is synthesized (block 8).

Product is then filtered (block 9), dissolved in a solvent and crystallized (block 10) before multidrop centrifugation removes the solvent (block 11) and final drying occurs in the (block 12).

3. 2.2.2 Plant lines description

Superplant

The plant size has been set as the number of the blocks in recipe times two, as no operations can be conducted in series (GMP regulations) and all operations can be performed in parallel, thus resulting in the minimal count of 24 superequipment units. Defining more superequipment units has no effect on results, it only slows-down the computation as the resulting designs contain 24 or less units.

Plant *C10*

Reactors: 8;	6.3 m^3: 1; 10 m^3: 5 ; 16 m^3: 2; each reactor has distillation column or condenser attached
Multidrop centrifuges:4;	0.44 m^3 each
Filters: 4;	0.6 m^3 each
Crystallizers:Ø	

Plant *Q2*

Reactors: 11;	4 m^3: 5, 6.3 m: 6; 3 reactors have distillation column or condenser attached
Multidrop centrifuges: 2;	0.8 m^3 each
Filters: 3;	0.6 m^3 each
Crystallizers: 2;	4.8 m^3 each

Plant *C11*

Reactors: 21,	4 m^3: 3; 6.3 m^3: 11; 10 m^3: 7 ; 12 reactors have distillation column or condenser attached
Multidrop centrifuges: 8;	0.44 m^3: 6; 0.8 m^3: 2
Filters: 9;	0.44 m^3 each
Crystallizers: 5;	10 m^3 each

In each plant line additional supporting units have been defined, e.g. dryers, tanks.

3. 2.2.3 Optimization settings

The first step includes setting–up the *Superplant* consisting only of superequipment units, selecting and specifying the target plants (*C10*, *Q2*, *C11*) in order to allow superequipment design performance computations in each of the plants and running the optimization with the following prioritized list of objective functions: 1. productivity, 2. special plant line selection score, 3. special diversification function, 4. Net Present Value, 5. number of equipment. The NPV refers to NPV of the campaign. In this case study no investment into additional or new equipment units is considered.

Later on, we optimized each of the plant lines (plant C10, plant Q2, plant C11) separately. Ordered objective function list was: special diversification function, productivity, floors–up indicator, net present value for the campaign, number of equipment.

Net present value of the campaign has been calculated as follows: customer placed an order of 750 t of Product H, projected selling price is 64 USD/kg of product. Plant independent costs (materials, waste treatment) are 30.4 mil. USD and energy costs, utilities, labour costs, plant line rent, changeover, overhead, other are considered plant and design dependent. Plant independent costs are 90 % of total costs on average. The interest rate is 10%.

The optimization settings for Tabu Search have been set as follows: neighbourhood size 40, the tabu list length was 90, forbidding shortly visited neighbourhoods. Stopping criteria for the optimization were set to 70 iterations after finding last non-dominated global optimum and there were 36 restarts during a run, because no new optimum could be found during the last 10 restarts.

Individual designs are created during the superplant optimization, where each of the units is represented by a superequipment unit. In order to obtain the productivity, costs and other parameters, real properties of each apparatus must be determined. The determination process is performed as many times as is the number of plant lines to compare, in our case three times for each design in each iteration. During the determination of unit properties, operation time, volume and mode have to be taken into account. This procedure results in a listing consisting of unit type, minimal size and minimal parameters required to carry out the desired operation block. Two final steps are then performed: identifying/matching the superequipment units with the real equipment from each plant line and scaling of the time and volume for each recipe block according to the bottleneck in order to obtain maximal productivity for each design in the three plant lines. The transformation of superequipment units follows in each iteration for each evaluated neighbor.

If the superequipment design can be matched into the given plant line, that means there is enough equipment in that plant line from each class, size and lining material for performing all operations. If the evaluated design contains more superequipment units than available units in a given plant line, either the design is not realizable in that plant, or additional investment is required. In such case, the maximal possible productivity is computed for the investment scenario. If no investment is allowed, productivity is zero.

3. 2.2.4 Results and discussion

The principle of matching the superequipment units from design #1 to a plant is demonstrated in Fig. 3-12. The second and third plant are matched the same way as the plant C10 and are not shown in the figure.

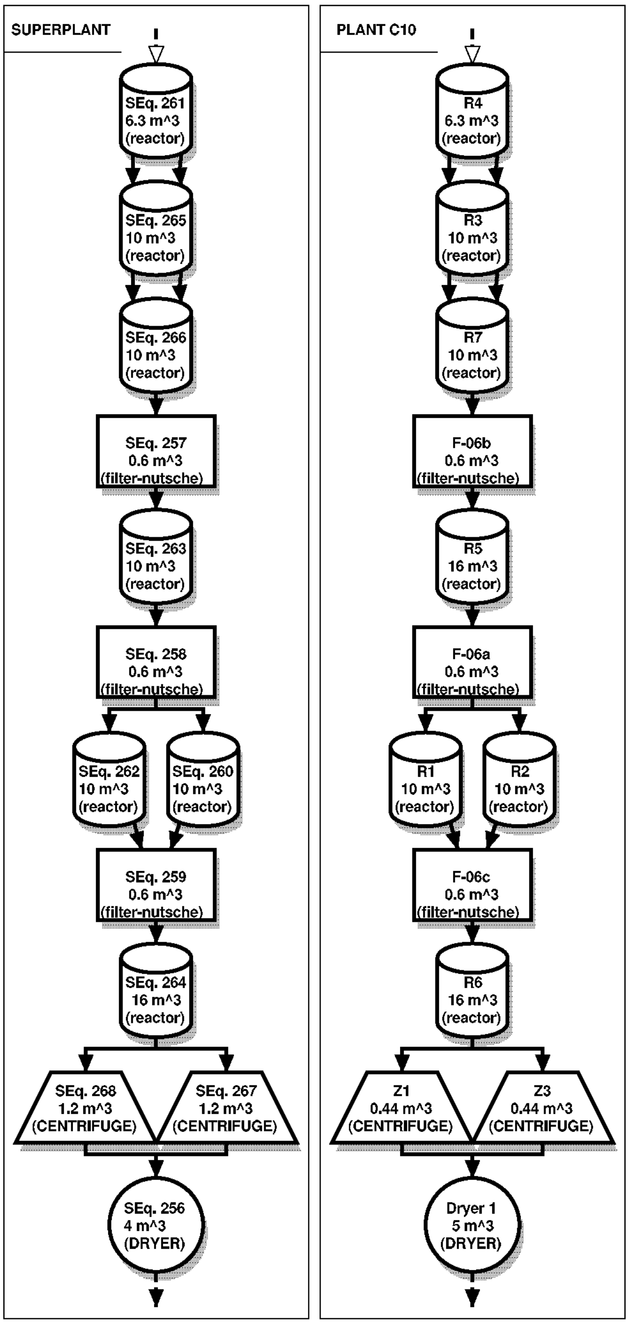


Figure 3-12: Product H case study. Design #1 as superplant layout (left) and mapped to plant C10 for evaluation purposes (right); (see Table 3-4). The superequipment unit volumes can be less than the equivalent volume in the existing plant. This can happen if the corresponding unit is not full. This is set according to the defined heuristics.

First, the selection of equipment sizes is constrained only by recipe block volume demand. For example, if the volume requirement of a reaction is 14 m^3, a superequipment unit obtains the next available nominal volume of a reactor: 16 m^3 as a first approximation in determining the volume of superequipment. Therefore the superequipment design #1 cannot be used directly for determining all the evaluation parameters in each selected plant. For example, because the standard size of centrifuge in the design #1 used in superplant is 1.2 m^3, the cycle time computed for this alternative is lower than the cycle time of the same design in the plant C10 using 0.44 m^3 centrifuges. Plant Q2 has no available 16 m^3 or

10 m^3 reactor, therefore the respective design #1 productivity will be determined by an appropriate superequipment layout transformation, where the SEq. 264, a 16 m^3 vessel becomes a volume bottleneck after transformed to a 6.3 m^3 reactor.

One of the most important questions for the decision maker is: "in which plant line should the campaign be carried out? ". Table 3-4 provides a comparison of parameters of designs mapped completely to the three plant lines (not requiring any investment, each design can be performed in a given unmodified plant line).

Table 3-4: Subset of resulting designs from *superplant optimization* for the case study Product H and the following plant lines: 1. plant C10, 2. plant Q2, 3. plant C11. Results are sorted according to productivity. The bold numbers indicate best designs in each plant according to productivity.

Design ID	Plant line ID	Prod. rate	Batch size	Nr. equip.	NPV
(#)	(#)	[kg/hr]	[t]	[pcs.]	[mio. USD]
3	3	135.1	1.27	24	23.0
7	3	133.0	1.24	21	22.8
1	1	130.9	1.60	13	23.3
4	1	130.9	1.60	14	23.2
1	3	90.3	0.97	13	22.1
2	2	74.0	0.69	13	21.0
4	2	68.8	0.63	14	20.6
1	2	58.4	0.63	13	19.3

As we see in Table 3-4, design #3 is the overall best in terms of productivity. This is also the best design found for plant *C11*. The large number of 24 units used for production implies increased costs for this design (NPV=23.0 mio. USD) and thus not reaching the optimum for the NPV. This design is actually performing all recipe blocks in parallel and thus creating overhead in cleaning costs and labour demand. The productivity per total nominal volume of significant equipment is low (0.71 $kg.hr^{-1}.m^{-3}$).

The best productivity in the plant C10 can be achieved by design #1 (Figure 3-12), being only 3% less effective than design #3, but it utilizes only 13 units, reducing the costs and reaching the optimum in NPV equal to 23.3 mio. USD for the whole campaign. Plant Q2 is dedicated for small productions and has no 10 m^3 reactor, therefore the best achievable productivity is 74.0 kg/hr with design #2. All of the resulting Net Present Values are rather similar, because of the high fraction of raw material and solvent costs. Thus the primary criteria for decision might be for example productivity, number of units used or batch size. If the campaign time is the main issue, design #3 should be implemented in plant C11. On the contrary, if the free capacity of a big plant (plant C11) was required for upcoming campaigns, design #1 in plant C10 could be a good compromise between performance, NPV and number of units used while keeping the large plant free.

Note that the same design can have multiple instances in the results table. For example design #1 from the superplant can be matched and realized in all of the plants without requiring additional investment. This means, that all 13 units from the superequipment design will be mapped into real units for each plant. However, because the equipment units differ in each plant, the performance indicators also differ and favour the second largest plant in the list, where the volume bottleneck is resolved by parallel usage of two 10 m^3 reactors.

Our goal is to search the whole solution space and obtain designs from minimal to maximal allowed number of units. Furthermore the specification declares, that only designs which fit into any of the real plants will be stored in the results list.

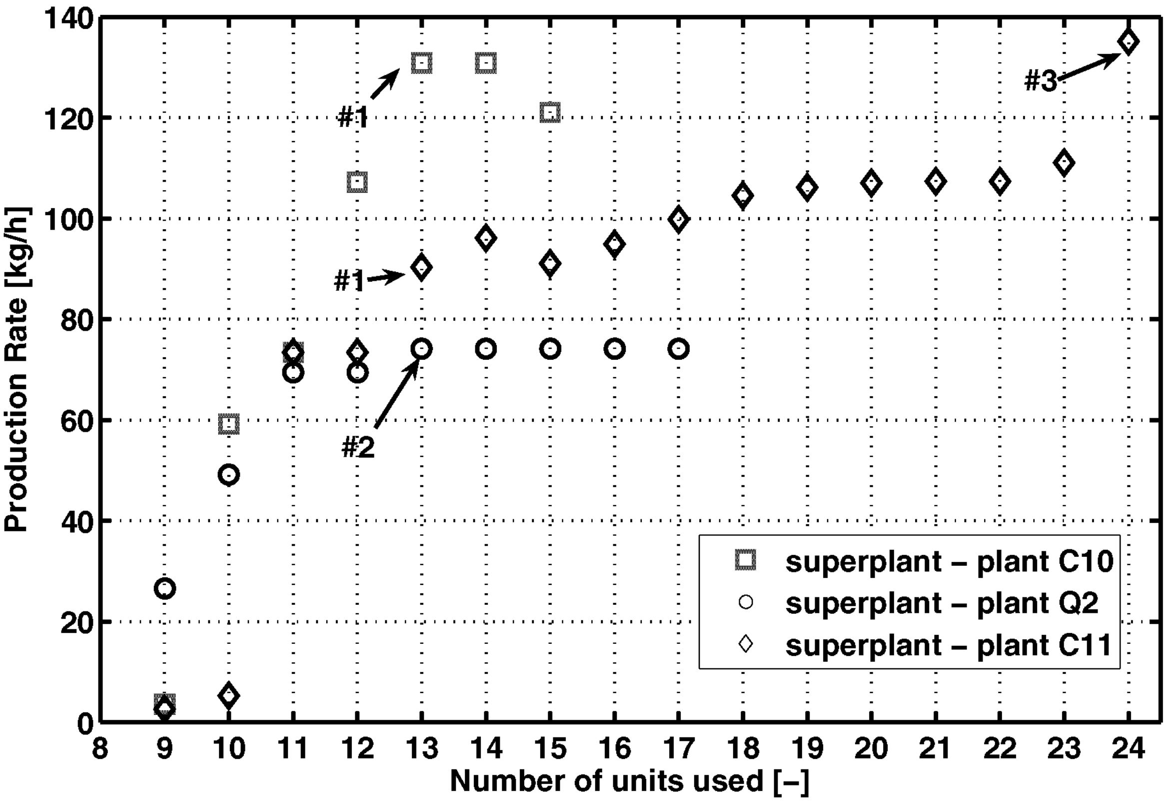


Figure 3-13: Productivity vs. number of equipment units for the Product H case study after matching superplant designs to plants C10, Q2, C11. Designs with maximal productivity for given number of equipment are displayed. Designs #1–3 refer to corresponding designs from Table 3-4.

The conventional optimization technique uses mostly restarts as the main diversification component. After each restart, a single random design is selected from the results list and random selection of move is not sufficient to escape the area with given number of equipment. Therefore the steering of searching algorithm in the objective space by special diversification function is needed, which ensures storing of designs with broad variety in number of units and a good productivity.

Figure 3-13 shows the relationships of the productivity vs. number of used equipment units for the three plants. A subset of designs with best productivity for given number of units and given plant is displayed. A minimal number of 9 units has to be used to process the 12 blocks of the recipe. The maximal number is 24 (design #3), as discussed above. The design #3 can be matched only with plant C11 without additional investment. The mapped designs show a trend of increasing productivity with increasing number of equipment. The best designs with 15 equipment units (Fig. 3-13) do not include two centrifuges in parallel mode (block 11) as in the best designs with 14 units. This increases the cycle time. If a design with 15 units and two parallel centrifuges (recipe block 11) is mapped into plant C10 or plant C11, an additional reactor (as compared to the best design with 14 units), which is placed in series with another reactor, becomes the volume bottleneck and no bigger unit is available. Similarly, in the plant Q2 adding equipment to the optimal design with 13 units improves the process no more.

Comparison of the results obtained by superequipment method with the standard optimization of each plant separately shows a good match (see Table 3-5). In the plant Q2, the non-dominated solution with 74.0 kg/hr and 13 units has not been found during the run (200 iterations without finding global optima and 40 restarts requiring ca. 230 minutes). This can be explained by too constrained searching space (fully loaded plant utilizes 14 units for this recipe) and neighbourhood, where the only possibility of diversification in current implementation of TS is in the restarts. On the other hand, superequipment method is less constrained due to the "chameleon" property of each unit and the diversification process is ensured naturally by moving through the solution space almost without limits (no restrictions in lining materials, size, etc.).

Table 3-5: Resulting designs with best productivity for conventional TS optimization of Product H case study in the plant lines: 1. plant C10, 2. plant Q2, 3. plant C11 (compare with Table 3-4 – superplant optimization).

Plant line ID (#)	Prod. rate [kg/hr]	Batch size [t]	Nr. equip. [pcs.]	NPV [mio. USD]
1	130.9	1.60	13	23.3
2	69.4	0.63	11	20.8
3	135.1	1.27	24	23.0

3. 2.2.5 Conclusions

Here, another advantage of superequipment concept becomes evident, as the freedom of equipment choice in each iteration is crossing the regions in the TS searching space without limitations. In the conventional method, the regions of invalid designs or infeasible solutions can be a limiting factor for escaping local optima and require other diversification methods for driving the optimization towards the global optima.

The CPU time analysis for standard TS optimization shows, that with increasing number of equipment units in the plant, the time to find global optimum increases. In the plant C10 optimization, global optimum in productivity has been found after 3900 iterations (mean optimization time 1040 s per 1000 iterations), in the plant Q2 after 5200 iterations (mean optimization time 1620 s per 1000 iterations), in the plant C11 after 8300 iterations (mean optimization time 1630 s per 1000 iterations). In total 26009 s were needed to optimize the three plants independently.

The superequipment plant optimization with 24 units and three lines required on average 3150 s per 1000 iterations and the mean of 8 runs on number of iterations to find global optimum in productivity for all three plants is 3600 corresponding to

11340 s. However the CPU time is strongly dependent on TS criteria selection, neighbourhood size, tabu tenure length, recipe complexity and objective functions selected. In any case the sum of optimization times for handling three lines separately is significantly larger than the optimization time utilizing the superequipment approach.

Again the superequipment concept offers a practical advantage: instead of specifying three independent optimization problems and obtaining three separate result lists as in the conventional approach, only a single problem has to be specified in the superequipment method delivering one overall list of results as a basis for decision making.

3. 2.3 New plant design for selected recipe - Acetylsalicylic Acid case study

The grass–root design is based on net present value calculations with investment and demonstrates suitability of superequipment concept on this problem type. The results of the superequipment concept optimization are compared with the standard plant line optimization in the grass–root simulation mode.

Figure 3-14 shows the grassroot design optimization scheme. A superstructure consisting of a number of superequipment units is defined as an input[7]. During a single optimization run, a number of designs consisting only of superequipment is generated. In each iteration all evaluated designs have to be "identified", which means transformation of superequipment model to an existing equipment unit takes place according to heuristic rules. The transformation process serves as a necessary step in determining the Net Present Value of each design.

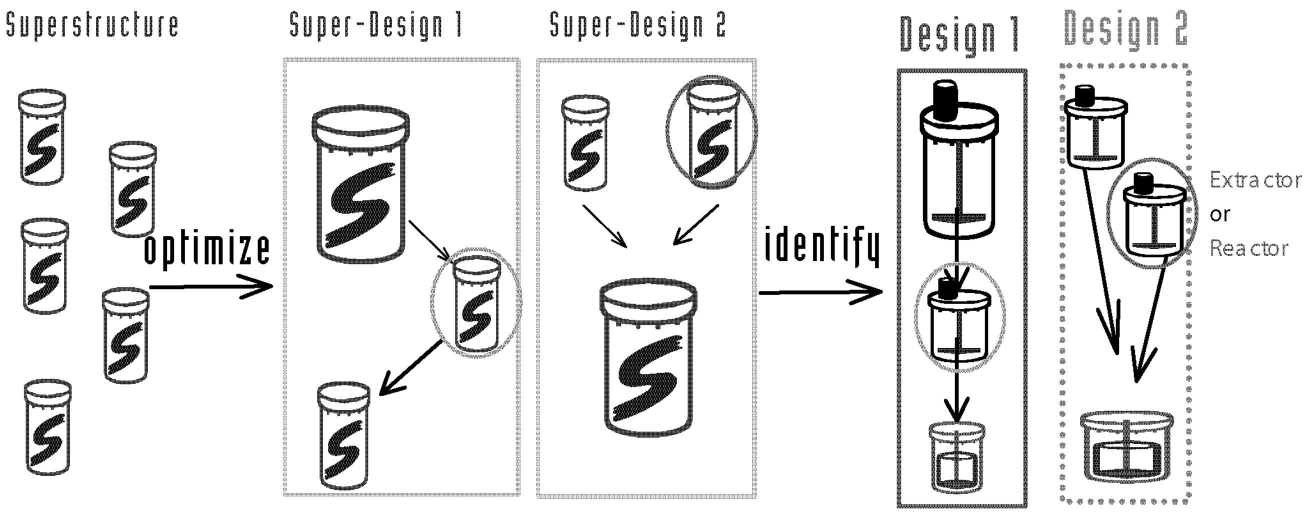


Figure 3-14: Grass-root plant line optimization scheme using superequipment. The superequipment units in a circle will be transformed for objective function evaluation into a real equipment unit, in order to determine productivity, NPV, and other objectives.

7. The superestructure domain definition is very simple, it is a number representing maximal number of potential equipment units in the new resulting grass-root plant line.

3. 2.3.1 Recipe description

Acetylsalicylic Acid is a low–cost fine chemical also known as Aspirin. In this case study a process first published by (Schmitt, 1885, Kolbe, 1860) and later reviewed by (Lindsey and Jeskey, 1957) is used. The appropriate reaction scheme is presented in Appendix A-5 *Acetylsalicylic acid reaction scheme, Kolbe-Schmitt synthesis*.

The process comprises 4 reactions, from which one is a reactive distillation. Reactants are charged and heated in a reactor. All of the solids dissolve and reaction takes place, after which the reaction mixture has to be cooled down (block 1). Extraction (block 2) and pH–adjustment follow (block 3). After that, distillation of the solvent is necessary (block 4). Charging and dissolving of additional reactant is assigned to (block 5). Preparation of reaction mixture by charging of solvents, reactants, dissolving them and adjusting pH (block 6) take place. Afterwards, the reactive distillation has to be conducted in a reactor with distillation column attached (block 7). Charging, reaction and cooling of the mixture follows (block 8). Crystallization (block 9) and multidrop centrifugation (block 10) recover the intermediate crystals. Final reaction step occurs (block 11) and aspirin product has to be purified in subsequent steps by crystallization (block 12), multidrop centrifugation (block 13), and drying (block 14).

3. 2.3.2 Plant lines description

Superplant

The superplant size is determined by the maximum number of units which can be used in a design. In this case, the recipe consists of 14 blocks, where only 4 blocks can be performed in parallel mode (constraint set in the recipe input) and no blocks are allowed in series for safety reasons. Therefore, the maximum number of units is 18, this is set as a size for superplant.

Plant C12 (Conventional TS optimization)

This plant consists of various types of reactors, dryers, tanks, multidrop centrifuges, filters, extractors, crystallizers and more units. Sufficient number of equipment classes to perform each block is available. We present here only the important equipment units. The plant was defined for comparison purposes with the Superplant. The basic assumption in this optimization case is that, if enough equipment units is provided for performing the recipe, the conventional TS optimization will automatically result in the appropriate grass-root designs.

Filters: 6; 0.6 m^3 each

Multidrop centrifuges: 8; 0.44 m^3 each

Dryers: 4; 10 m^3 each

Crystallizers: 4; 10 m^3 each, PTFE lining material

The recipe can be performed by a maximum of 12 reactors (reaction, crystallization, extraction, pH adjustment, distillation operations, etc. can utilize a reactor) in 11 blocks, where one of them can be performed in parallel. Standard temperature and pressure conditions used in the recipe require common vessel materials up to 6 bar and 200 C. We consider only three standard sizes for reactor vessels: 16 m^3, 10 m^3 and 6.3 m^3. We also consider three possible lining materials in the reactor vessels: stainless steel, Hastelloy, enamel/PTFE. All possible combinations of such properties result in 216 reactor units in the grassroot plant.

As we want to simulate a grass–root design with this plant, we neglect the floors, spacial coordinates of each unit in the plant and also the connectivity constraints.

3. 2.3.3 Optimization settings

For the grassroot optimization of superplant and plant C12, the prioritized list of objective functions has been set as follows: 1. production rate, 2. NPV, 3. costs of campaign, 4. special diversification function, 5. number of units in design.

The main interest of the grassroot design is the calculation of economic indicators (planning phase of a project), which are included as objective functions: NPV and costs of the campaign. The costs of the campaign are needed as an input for the NPV computation. The plan is to produce Aspirin during the next 5 years. It is assumed that the whole production can be sold during this period. The projected selling price used in computations is set to 8.76 USD/kg of Aspirin and remains constant during the production period. The equipment is bought before the beginning of the campaign and is computed for each design individually. Installation factor, material type, pressure factors, options, etc. are also included in the pricing. Costs of buildings, building area and costs related to the beginning phase of the project are not included, as we consider only the costs directly related to the design. The loan begins one year before the start of the production and the interest rate is 10%.

The costs of the campaign include labour, utilities, changeover, materials and waste costs, overhead and other. The plant independent costs include mostly the material and waste costs and represent about 75% of the total costs without investment for this campaign. Cash flows are computed at the end of each year. The equipment units are to be depreciated during the 5 years period from the beginning of production; the salvage value is set to zero.

The Tabu Search options are set as follows: neighbourhood size: 40, tabu list length: 80, number of iterations until restart without finding new non-dominated solution: 80, number of restarts: 30.

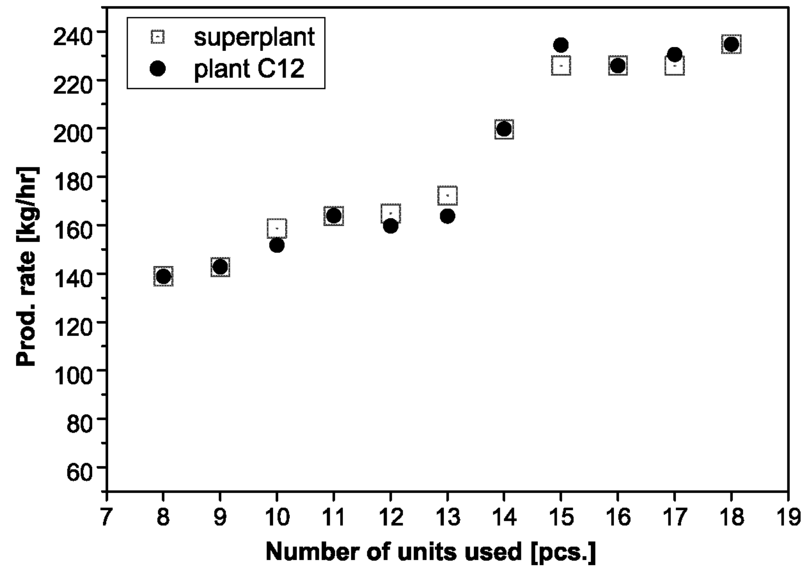
3. 2.3.4 Results and discussion

The TS optimization shows that the minimal number of units needed for the production is 8, the maximal is 18 due to the recipe specific heuristics. Table 3-6 displays three representative designs: Design #2 is an optimum in productivity, Design #14 is optimum in NPV and Design #407 is utilizing only 8 equipment units. All of the designs have a payback time of less than five years.

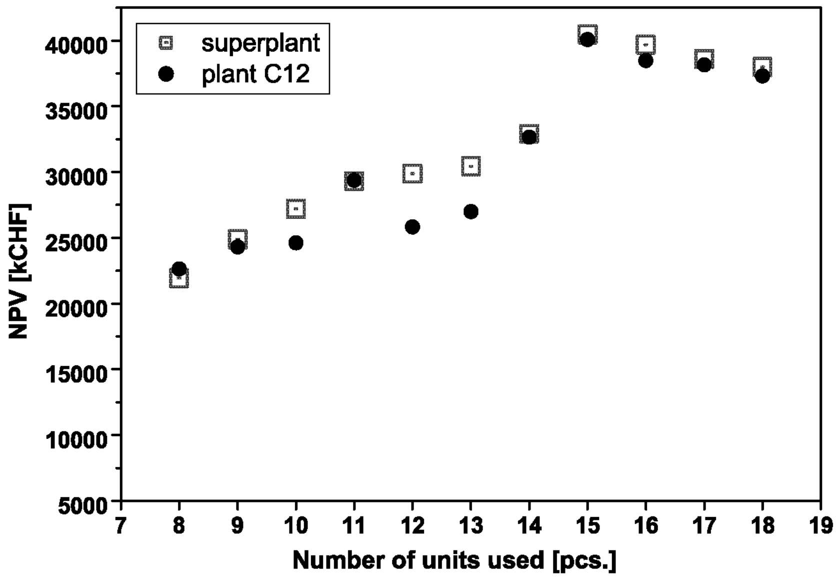
Table 3-6: Aspirin case study: subset of dominating (approximated Pareto–optimal) designs for the grassroot problem as obtained by superequipment approach.

Design ID (#)	Prod. rate [kg/hr]	Batch size [t]	Nr. equip. [pcs.]	NPV [mio. CHF]
2	234.7	2.24	18	38.1
14	226.4	2.24	15	40.5
407	137.1	2.09	8	24.1

Figure 3-15a shows a graph of maximal productivity vs. number of units used in a design as stored in the results list.



(a)



(b)

Figure 3-15: Aspirin case study. Resulting designs from superplant optimization (hollow squares) and plant C12 optimization (dark dots), maximal productivity vs. number of units (a) and maximal NPV vs. number of units used (b).

It includes a subset of dominating designs according to the objective functions list. The dominating design #2 utilizing 18 equipment units from the superplant optimization has the best productivity of 234.7 kg/hr. The same level of productivity has been identified with the conventional optimization of plant C12 , where the structure of transfers between the blocks in the optimal designs is comparable.

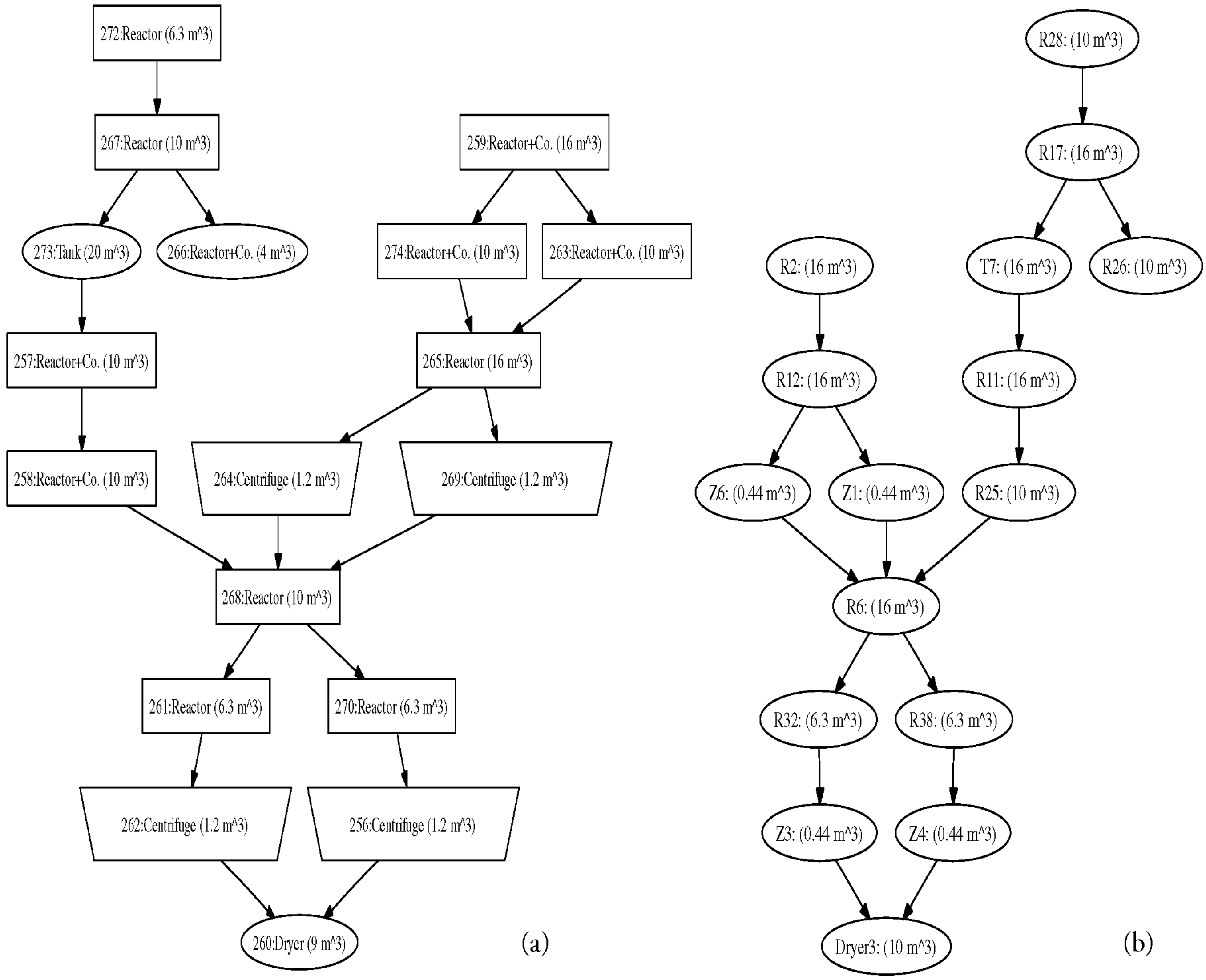


Figure 3-16: Aspirin case study. (a) Design #9, superplant (superequipment concept optimization) (b) Design #1 plant C12 (conventional TS optimization). Z denotes a centrifuge, R a reactor, T a Tank, Co. is a condenser. The ellipses represent conventional equipment, the rectangles represent transformed superequipment units.

The scatter plot of NPV vs. number of equipment units displayed in Figure 3-16 (b) shows a clear dominance (approximated Pareto–optimum) with a design utilizing 15 equipment units (the tanks do not count to the total number of equipment). The NPV value for this case is 40.5 mio. CHF. It has to be mentioned again, that the superequipment method cost algorithm prefers the cheapest possible unit if a selection for a given recipe block is available.

Figure 3-16 shows, that the TS optimization using superequipment method is capable of finding the same structures within the designs as in the conventional

TS optimization method. Both designs use 2 pairs of centrifuges in appropriate recipe blocks and all equipment locations and sizes are comparable. The unit sizes in superequipment design must be equal or larger than the corresponding units in plant C12 design in order to achieve the same productivity. Otherwise, the superequipment design productivity is scaled-down according to the volume bottleneck.

3. 2.3.5 Conclusions

In the superplant method the optimal productivity has been found after 4200 iterations requiring ca. 3350 s per 1000 iterations (14070 s in total). The conventional TS method requires a mean value of 5680 s per 1000 iterations and 6400 iterations to find global optimum (36352 s in total). The plant definition for the Plant C12 comprises more than 220 units, which is a burden for CPU time as compared to only 18 superequipment vessels.

The conventional TS optimization uses explicit specification of each unit, which might enable only suboptimal solutions when even a large number of specified units results in a too limited superstructure. This problem is overcome by optimizing only on the transfers, i.e. by using the superequipment concept for which only the maximum number of units has to be specified.

3. 3 Discussion and conclusions

We presented a novel approach using superequipment concept for solving different design problems related to single products in multipurpose batch plants: additional investment into an existing line or plant, the selection of one production line out of several available, and grassroot design of a new plant. Multiple objectives are considered in these design problems. Dominating (approximated Pareto–optimal) solutions are generated by means of a Tabu Search algorithm.

In the novel approach the concept of superequipment has been defined as an abstract model, which is capable of performing any physico–chemical operation. Each superequipment is transformed into a real equipment unit, for example a reactor, during or after the optimization in order to evaluate performance parameters of a design. This novel concept uses an implicit definition of a superstructure and essentially optimizes on the transfers between different equipment units. This principle perfectly mirrors the major aim of batch process design, i.e. the goal to equally distribute the tasks to be conducted to a number of equipment units so that the cycle time is minimized and optimal batch size is achieved.

On the basis of the case studies we demonstrated that the application of the superequipment concept has the following advantages for the investigated design problems:

In the superequipment mode, only the maximum number of additional units, each represented as a superequipment unit, has to be specified instead of defining a large number of equipment explicitely.

The multiobjective Tabu Search algorithm generates a number of superequipment designs in a single run. Afterwards the potential implementation of these designs into the available production facilities is investigated and the resulting performance indicators are determined. The indicators such as productivity and NPV highlight the trade–offs between different facilities. Constraints such as due dates can be considered in the selection of the most appropriate line.

The comparison with optimization results obtained with a conventional Tabu Search algorithm revealed that the superequipment approach is capable of identifying the dominating solutions. CPU time comparison of conventional TS optimization vs. superequipment TS optimization shows that the superequipment method requires in some cases more time per iteration (2.1 s/iteration for the Vitamin C investment case compared to 0.9 s/iteration), which is due to additional heuristics and increased time for evaluating objective functions for all options in one superequipment. However, the number of iterations to find global optimum is lower than in the corresponding traditional TS optimization case, because the search space and move space are less constrained. For the investigated case studies the overall CPU time for the superequipment approach was significantly lower as compared to the conventional approach.

For all three fields of application the superequipment concept means a considerable saving in time and effort because the optimization problems are reduced in size and repetitive optimization runs are avoided.

Other possible applications include, but are not limited to, optimizing of several given plant lines with possible constrained investment options simultaneously, finding a set of designs for given number of recipes in one optimization run, optimization in an equipment pool plant with several recipes and layout combinations. It has to be investigated to which extent the superequipment concept can also be used in combination with other optimization approaches.

Uncertainty with application to design robustness measures

4

Abstract

This chapter adduces multiobjective optimization with uncertainty based on approximated Pareto-analysis and exemplifies the principles on a multipurpose batch plant study.

There exists a tendency of incomplete information in the early industrial batch process development. The unknown variables are often related to a preliminary chemical recipe, for instance duration of operations. The aim of the preliminary batch process design is to find assignments of recipe operations to given equipment units available for production, which we call the design or solution. Due to the presence of uncertain factors in the early stages, quality of these assignments is often difficult to quantify. Similarly, if the goal is to obtain the best design for implementation in the multipurpose factory, the quantification of objective functions needs also to be adjusted for uncertain variables.

We propose a Tabu Search optimization framework, which allows to find a set of feasible designs (dominating optima, local optima and interesting solutions) with uncertain variables present in the initial recipe. As a novel approach, we introduce productivity of a design [kg/hr] as primary and productivity robustness as a secondary objective function, which is an important measure of design reliability under varying conditions.

This combination of objectives is able of providing results list based on Latin Hypercube Monte Carlo simulation consisting of guaranteed valid designs. The implementation of Tabu-Search framework using a two-stage programming approach and uncertain process input parameters results in huge flexibility in the input stage of problem specification. Adding and modifying of objective functions is inherited.

As a result we successfully demonstrate that meta-heuristic optimization techniques are capable of delivering feasible solutions focusing on optimizations of performance robustness on a large domain of process designs and are also able of capturing different antagonistic solution qualities by multiobjective optimization.

4. 1 Introduction

In the Chapter 2 *Methods & algorithms*, we introduced the methods needed for searching for optimal batch process design. In this chapter, we describe two case studies: a synthetic process with an arbitrary recipe and a simple Aspirin production, in which we evaluate the uncertainty and the robustness of the resulting designs. The aims of the case studies are:

- to find a set of batch design alternatives with good performance and outstanding robustness
- to perform a monovariate sensitivity analysis of the uncertain recipe data for each resulting design
- to estimate changes in productivity according to the multivariate uncertainties in the initial recipe for each resulting design

4. 2 Uncertainty case study

4. 2.1 Initial recipe

It is a hypothetical production in preliminary (R&D) phase, where only limited knowledge about the process is available. The recipe tasks should mimic typical operations found in a speciality chemical production. In this stage, we assume that the reaction data is not fully determined, but an approximate basic mass-balance can be computed prior to the optimization, which we use as a base case for scaling-up of the production.

The recipe displayed in Table 4-1 contains 15 steps, from which three are reactions. It is a production-scale process performed by equipment of a base volume of 10 m^3 and assuming a simple linear-arrangement of all equipment units necessary for successfully performing the production.

Table 4-1: The uncertainty case study, an arbitrary recipe with uncertain operation times. The functions trirnd() and normrnd() refer to random triangular and normal PDFs.

ID	Operation description	Time [min]	Volume [m^3]	Operation class	Flags	Temperature [°C]	Pressure [bar]	Lining material	Previous operation	Next operation
1	1.0 Charge	20	8.4	4	0	20	1.0	-1	-	2
2	2.0 React	trirnd(230, 350, 400)	8.4	32	1	180	4.0	-1	1	3
3	3.0 Charge	20	9.4	4	0	90	1.0	-1	2	4
4	4.0 Distill	160	9.4	17	2	80	0.5	-1	3	5
5	5.0 Charge	20	2.6	4	0	80	1.0	-1	4	6
6	6.0 Crystallize	210	2.6	15	2	50	1.0	-1	5	7
7	7.0 Multidrop centrifuge	200	2.5	47	2	30	1.0	-1	6	8
8	8.0 Charge	20	3.1	4	0	20	1.0	-1	7	9
9	9.0 React	trirnd(300, 400, 500)	3.1	32	2	100	4.0	5	8	10
10	10.0 Extract	120	3.6	19	2	50	1.0	-1	9	11
11	11.0 Charge	80	3	4	0	50	1.0	-1	10	12
12	12.0 React	normrnd(420, 16.80)	3	32	0	100	4.0	-1	11	13
13	13.0 Crystal-lize	250	3	15	2	50	1.0	-1	12	14
14	14.0 Multidrop centrifuge	300	1.5	47	2	30	1.0	-1	13	15
15	15. Dry	200	0.5	18	3	90	1.0	-1	14	-

The time of each operation is set as fixed or uncertain. The three reactions: *Reaction* 2.0, 9.0, 12.0 contain uncertain time data in a form of probability distribution functions. The first two reaction durations utilize the triangular Probability Density Function (PDF), which is usually used if expected and minimal/maximal durations are available. The triangular distribution, if known from the process, is providing more information than the *min–max* approach. The third reaction with a time set to *normrnd(420, 16.80)* PDF is taking 420 minutes on average, and is expected to follow the normal distribution with a standard deviation $\sigma = 16.8$. Such distribution of reaction time could be a result of more detailed laboratory measurements. The volume of operation is a maximum-filling volume of an initial production-scale equipment. Although operation volume can be considered uncertain in this phase, this option is not investigated here. A changing volume in the initial recipe could simulate for example the varying amount of the reaction mixture. *Operation class* refers to the class of operation listed in *Operation description* column.

The *Flags* column in Table 4-1 represents constraints to the optimization problem: 0 for no constraints, 1 for parallel mode of operation forbidden for given operation, 2 for in-series mode of operation forbidden for current operation, 3 for in series and in-parallel mode of operation forbidden for current operation. If more operations are to be conducted in the same equipment unit, the constraints add up. The temperature and pressure refer to maximum temperature and pressure reached during a given operation. Lining material is a constraint, which assures usage of proper equipment lining material type for each operation, for example, if a fluorinating reaction was used, lining material PTFE (with a numeric representation 5, listed in Section 2. 3.2 *Options, constraints and limitations*) would be used. In this simple example, all lining materials except one are non-constrained (and thus set to -1).

The columns *Previous operation* and *Next operation* represent a link to previous/next operations which is important for non-linear recipe types, for instance if two reactants are to be prepared simultaneously (two parallel branches in the recipe) before mixing in one common reactor (final reaction branch of a recipe tree). The listed recipe is linear in the operation arrangement.

4. 2.2 Plant line

The multipurpose batch plant considered in this case study (see Table 4-2) contains 8 reactors of different sizes, multidrop-centrifuges, extractors and a dryer. Additional supporting equipment as pumps, heat-exchangers, mills, etc. is not listed. Each equipment unit is characterized by its class, class number, nominal volume, floor (eventually coordinates in the building), lining material and TP range.

Table 4-2: Plant line used in the uncertainty case study.

ID	Equipment name	Equipment class name	Equipment class ID	Nominal volume [m^3]	Floor	Lining material	TP range
1	Ce1.6a2	Centrifuge	135	1.6	0	5	1
2	Ce1.6b4	Centrifuge	135	1.6	0	5	1
3	Cr10a2	Crystallizer	5	10	1	5	1
4	Cr10a2	Crystallizer	5	10	1	5	1
5	Cr10b4	Crystallizer	5	10	1	5	1
6	Dryer1	Dryer	16	6.3	1	5	1
7	Extract1	Extractor	6	4	1	1	1
8	Extract2	Extractor	6	4	1	5	1
9	R10c1	Reactor	14	10	3	5	1
10	R4a1	Reactor	14	4	3	5	1
11	R4e1	Reactor	14	4	3	5	1
12	R6a1	Reactor	14	6.3	2	5	1
13	R6a2	Reactor	14	6.3	1	5	1
14	R6b3	Reactor	14	6.3	2	5	1
15	R6d4	Reactor	14	6.3	2	5	1
16	R6f1	Reactor	14	6.3	2	1	1
17	T10a2	Tank	19	10	0	1	1
18	T16a3	Tank	19	16	0	1	1

4. 2.3 Optimization settings

A standard Tabu-search method of optimization was used for this case study. Prioritized list of objective functions was: 1. productivity, 2. robustness. All three robustness objectives were examined in three separate runs. Robustness #3 was computed according to Eq. 32 with parameters: $c_+=0.25$; $c_-=0.25$ (see Section 2. 3.4).

For data sampling, a Latin Hypercube method (Hess et al., 2004, Saliby and Pacheco, 2002) resulting in 300 points was used, 100 points per each uncertain variable so, that each variable is tested in its whole range. Additionally for monovariate sensitivity analysis another 36 data sampling point combinations were defined, 12 for each variable. The random numbers were generated according to standard probability distribution functions (Equations 33 – 35). The data sampling procedure is repeated for each evaluated neighbour from the solution neighbourhood before the actual evaluation of productivity takes place.

In the second stage, the resulting 300 discrete productivity values are fed to the robustness objective function. As a result, the objective function value of robustness as a scalar is obtained for given solution according to Figure 2-15 and Equations 29 – 32.

For the TS optimization aspiration criterion was used, a short term Tabu list of a length of 12 and a stopping criteria which would restart the run for 10 times if no improving moves to the global optimum were recorded successively for more than 30 times.

The neighbourhood of the current solution was evaluated to a degree ranging from 20 % to 40 % and a candidate was selected according to TS heuristic rules. If the neighbourhood was smaller than 12, all neighbours were selected for evaluation. During the run, due to the large number of combinations of equipment and moves, a standard average neighbourhood size of 80 members was encountered.

Storing of the resulting designs was following the priority list: 1. global optimum, 2. local optima in one of the objective functions, 3. approximated Pareto optima (dominating solutions), 4. other good performing designs with a limit of 75% of the value of the global optimum.

4. 2.4 Case study results

The results list, consisting of about 30 designs with optimal and good-performing objective function values, is filtered to display the four most interesting solutions (see Figure 4-1), which contains:

- Design #1, which is an optimum in the expected productivity value (which is a value with the highest probability in the resulting discrete PDF) and also an optimum in the maximum productivity value from all the results
- Design #6, which is an optimum in the robustness R_1, R_2, R_3
- Designs #3 and #7, which are dominating (approximated Pareto optimal)

Please note, that the design identification numbers do not refer to any particular sorting order, they serve only for naming purposes. Each design on Figure 4-1 contains a histogram of productivity and a lognormal fit of the productivity data. The required objective function definitions are given in the Section2. 3.4 *Robustness measures*.

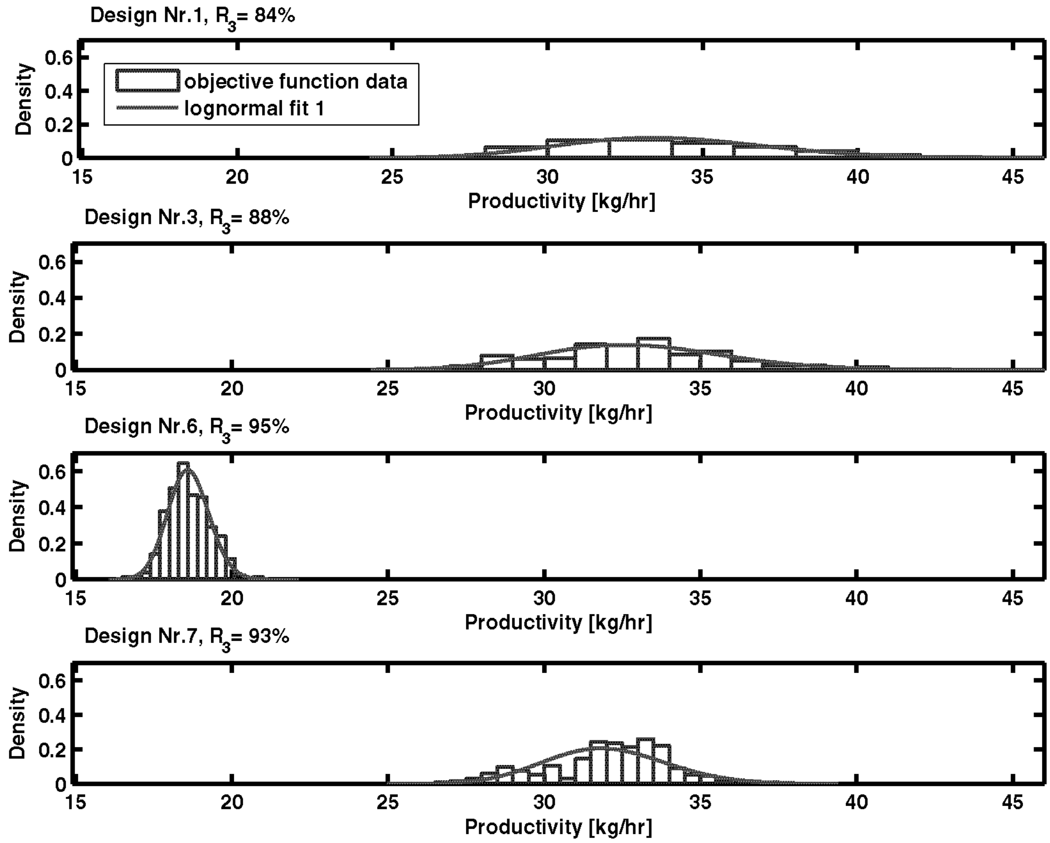


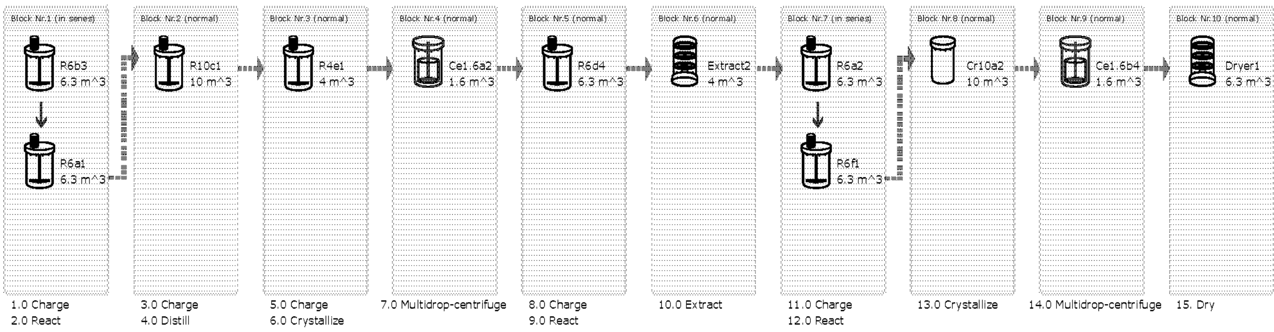
Figure 4-1: Results list for the uncertainty case study depicting probability density distribution vs. productivity of four designs. Solution #1 is optimal in the expected productivity, the design #6 is optimal in robustness and other designs are dominating (approximated Pareto optimal). Robustness #3 indicator is displayed in the header of each graph.

After reviewing the results list, the decision maker will select a few designs for further inspection. If the most important decision criterion was robustness, one might select the design #6, because it shows (a) high probability of the expected productivity value; Robustness #1 = 62.0% (b) a high probability of achieving the expected productivity or higher; Robustness #2 = 58.6% and (c) a high Robustness #3 value = 94.9%, which means that the span between the expected productivity and the expected productivity ±25% is 5.1 %. In other words, the most probable operation mode ($cdf(G_{max})$ ±25%) among the examined probabilistic states lies within the 5.1 % deviation interval from the value G_{max} = 18.2 kg/hr. However, this design provides only low productivity values.
Therefore if the decision maker aims at high productivity, one might select Designs #1 and #7 for inspection, where we can see differences in the assignment of recipe operations to equipment units (Figure 4-2). The Robustness #3 values are 84 %, resp. 93 %. The former design provides higher expected productivity and higher maximal achievable productivity and the latter only a slightly lower productivity compared to the optimum with higher Robustness #3 indicator.

Design #3 is dominated, achieves an expected productivity value of 32.2 kg/hr and Robustness #3 indicator is 88 %. The Design #3 will not be further investigated.

(a)Design (#1) flowchart.

Production Rate [kg/h]=33
Number of units used [-]=12
Robustness #3 [%]=84



(b)Design (#7) flowchart.

Production Rate [kg/h]=32
Number of units used [-]=11
Robustness #3 [%]=93

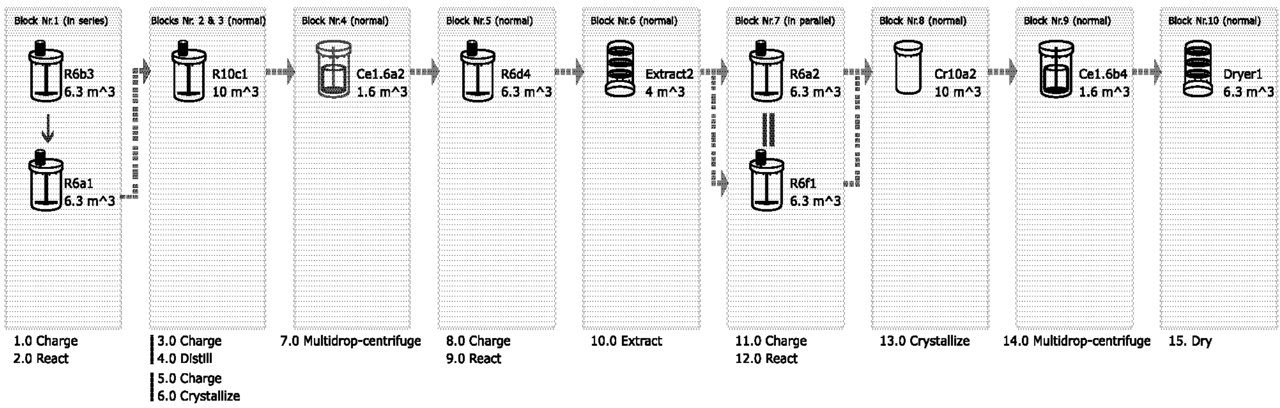


Figure 4-2: Uncertainty case study results: (a) Design #1 and (b) Design #7 flowsheets. The equipment units correspond to the factory units listed in the Table [4-2]. The vertical arrows mean in series mode of operation, the two parallel vertical bars between units mean in parallel mode of operation. Two or more blocks can be performed by the same equipment (see Design #7, tasks 3.0 — 6.0). Robustness #3 value is displayed in the header along with other objective function values.

Figure 4-2 shows that the Block nr. 7 is performed in series in two reactors for the solution #1. This sequence of reactors shortened the cycle time compared to the initial design. The second solution, Design #7 shows two reactors in parallel mode of operation for the operation 12.0 *React*, which improves the batch size. However, for the volume demanding operations in the first block (operations 1.0 and 2.0), both designs utilize two reactors in series, which will shorten the operation time, but on the other hand, both reactors are only 6 m^3, which creates a volume bottleneck.

As can be seen from Table 4-3, indeed, all the reactors used for the first block are volume limiting (reactors R6a1, R6b3). We can also conclude, that the latter design could be improved substantially in the batch size by using larger reactors for the first block, which is not realizable in the current plant.

Table 4-3: Volume filling grade for Design #1 and Design #7 from the uncertainty case study. The centrifuges are never volume limiting.

Equipment unit name	Volume filling grade Design #1	Volume filling grade Design #7
Ce1.6a2	-	-
Ce1.6b4	-	-
Cr10a2	0.225	0.225
R10c1	0.705	0.195
R6b3	1.000	1.000
R6d4	0.369	0.369
R6a2	0.357	0.179
Extract2	0.675	0.675
Dryer1	0.024	0.024
R6a1	1.000	1.000
R6f1	0.357	0.179
R4e1	0.488	--

The cycle time of Design #1 is (min., average, max.): 175 min, 226 min, 276 min and the cycle time of Design #7 is (min, average, max): 199 min, 232 min 277 min. Figure 4-3 shows the Gantt-chart of the Design #1 and #7, a utilization of equipment units over time. This is an average operation time representation. In the worst case, all uncertain operations are prolonged, in the best case all relevant operations will be shorter according to the initial recipe specifications. The first design‘s cycle time is determined by operations 8.0, 9.0, the second design‘s cycle time by operations 11.0 and 12.0 performed in parallel mode. Because the time limiting operations are different in comparison of the two solutions, it can be assumed that the robustness of the designs will differ.

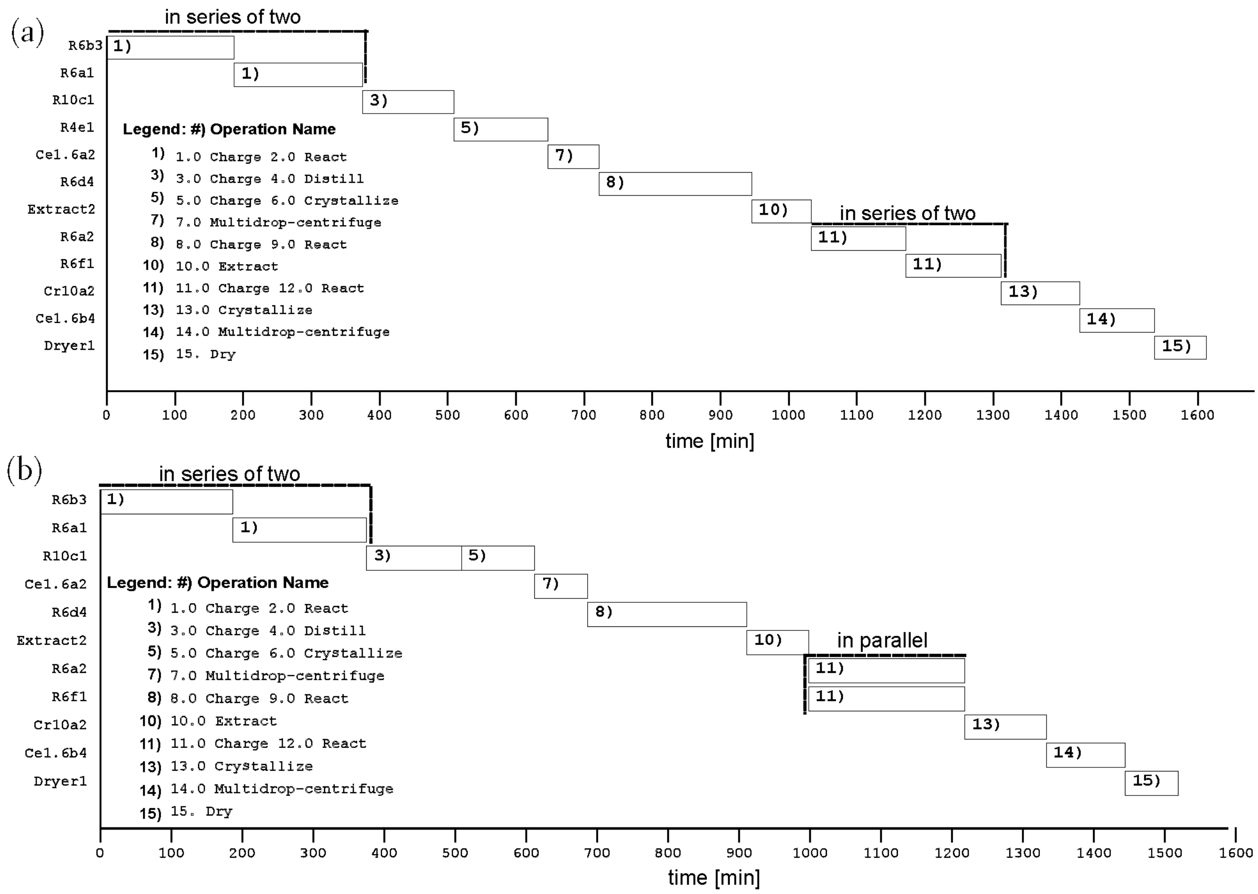


Figure 4-3: Uncertainty case study results: (a) Design #1 and (b) Design #7 Gantt-charts. Marked are the in series and in parallel operations. Time of the operations represents the most probable time among the number of uncertain states.

If we look at the sensitivity analysis on Figure 4-4, which was computed as a response in productivity to a single variable modification, where all other uncertain variables have been fixed, we can identify similarities in the dependencies. For the Reaction 12.0, the productivity of Design #1 is insensitive according to this analysis, and Design #7 shows lower productivity if the reaction time is too high. This is in agreement with the in series (Design #1 where this is not a time limiting operation), resp. in parallel (Design #7, operation is time limiting) mode of operation for this reaction. However, the correlations between the three uncertain reaction times cannot be seen.

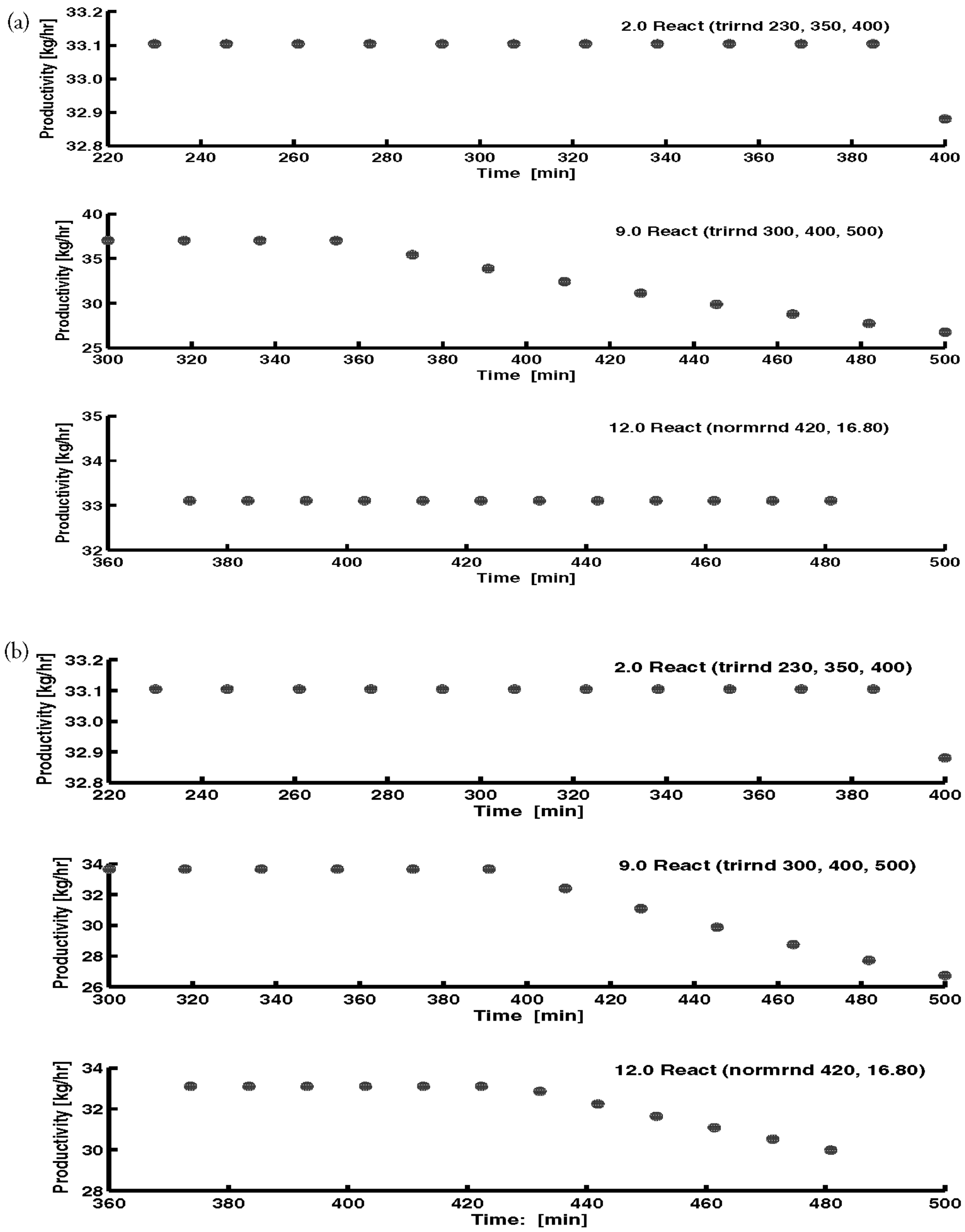


Figure 4-4: Uncertainty case study, monovariate sensitivity analysis for (a) Design #1 (b) Design #7.

Figure 4-5 shows a multivariate analysis of the productivity–reaction time dependence with probabilities. The data points represent a histogram of probabilistic areas, where the dark colors close to black mark the most probable states. Light colors close to white represent a probability close to zero, which means that such state is improbable. Several basic shapes can be observed, for example linear shape (case (a), second graph) means that the productivity is highly correlated to the *Reaction* 9.0 time variable with the most probable coordinates around the [32.7 kg/hr, 400 min] point. The correlation becomes smaller if the *Reaction* 9.0 time is lower than 350 min, because other factors begin to influence the productivity. One of the most important results for R&D stage here is: if the design #1 is going to be implemented, there will be a strong correlation between reaction 9.0 time and productivity. However, lowering the reaction time under 350 minutes will not bring further increase in productivity. This effect can be observed even better for the Design #7 (Figure 4-5 (b), second graph from bottom), where lowering of the reaction 9.0 time under 400 min is not going to improve the productivity dramatically, because the influence of other uncertain variables is too strong.

An elliptic shape of the histogram, as seen in Figure 4-5a - reaction 12.0 and Figure 4-5b - *Reaction 2.0* means that the correlation between the uncertain variable and productivity is close to zero. Of course, this information is design-dependent. Therefore robustness investigations conducted in early stages of batch process design can help to reduce research cost by pointing out relevant uncertain variables that need further investigation in order to identify optimal operating conditions in combination with a selected process layout.

Figure 4-5 (b) presents another important shape for the *Reaction 12.0*, which is a transition between an elliptic and linear shape, that means that the data is partially correlated to the productivity of the given design. If the reaction time will be kept between 400 and 450 minutes, there is a high probability of achieving productivities between 32–34 kg/hr. However, the histogram shows also, that the productivity can drop even under 28 kg/hr. There is a clear time-bottleneck front visible on the high-probability border.

Comparison of two designs based on presented histograms will show also the degree of productivity robustness regardless of robustness definition. The concept of robustness #1 can be observed and confirmed by the relative probability of the highest peak in the graph, that means the darkest color point. The higher the relative probability of a given state for all uncertain variables, the more probable the state in which the design will operate. Robustness #2 and #3 can be associated with the surface area of the histogram for each uncertain variable. The smaller the surface, the more precise definition of the production state. That shows also which and how important the input variables are in relation to specific design and productivity. The crosses in the graphs determine the expected productivity state and the expected operating times for the uncertain operations. In the probabilistic state indicated by these cross marks, the uncertain operation times are set to the basic operating conditions: 2.0 React: 350 min, 9.0 React: 400 min, 12 React: 420 min, which is equivalent to the base case recipe without uncertain times. From these operation points, the expected productivity value is computed, which is used for determining if the design is dominating.

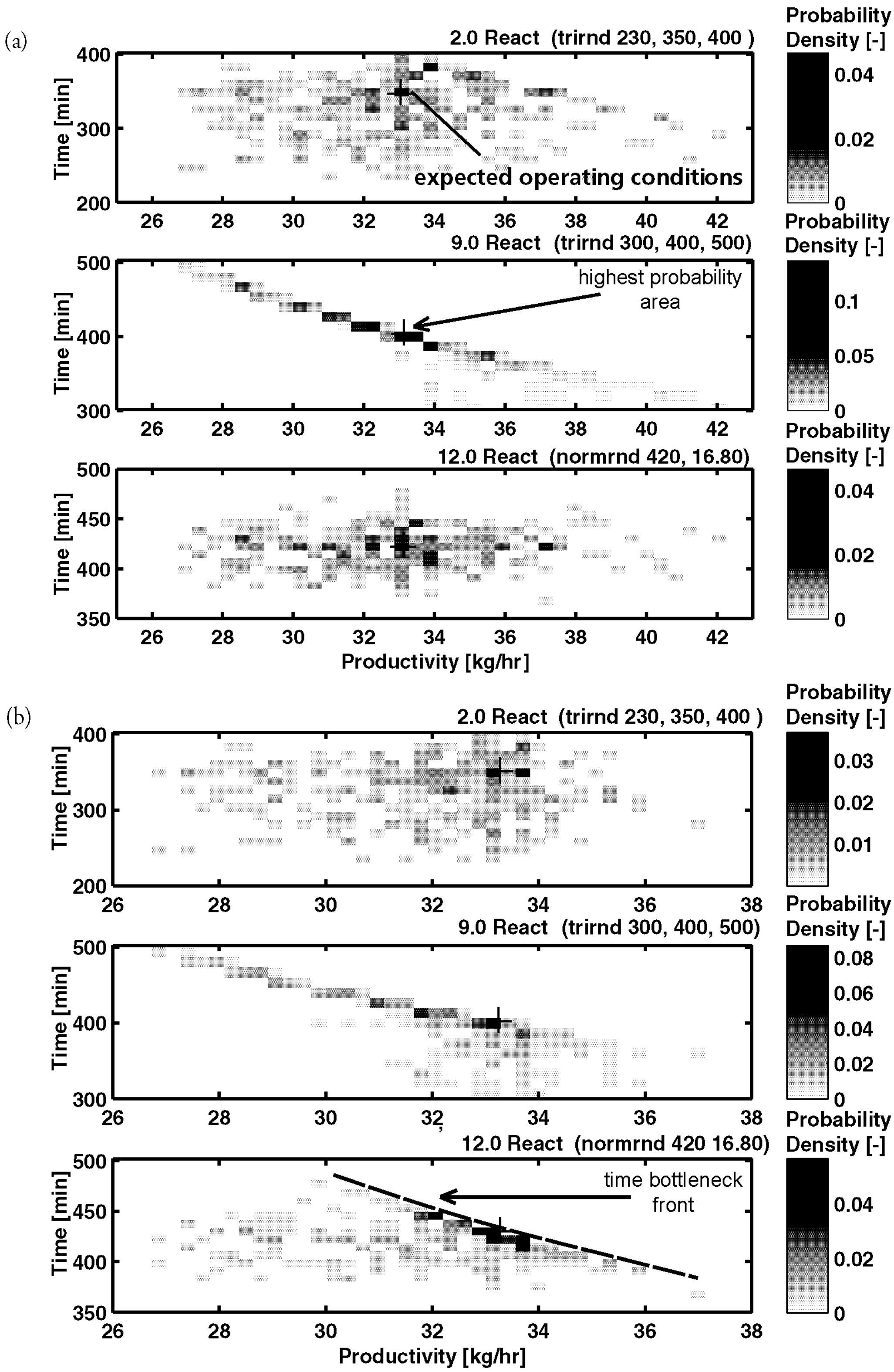


Figure 4-5: Uncertainty case study; multivariate sensitivity analysis, as a result of Latin Hypercube Monte-Carlo simulation with 300 points for (a) Design #1 and (b) Design #7. The Y-axis refers to the uncertain input variables in the recipe. The *normrnd*(μ, σ) function is synonymous to the to random number generator with normal distribution. The colored bars on the right-hand side refer to the relative probability density of the depicted rectangle areas in the graph. The cross marks refer to the expected operating times and expected productivity.

4. 3 Aspirin case study with uncertainty

Acetylsalicylic Acid is a low cost fine chemical also known as Aspirin. The appropriate Batch Plus recipe is listed in the Appendix A-6 *Acetylsalicylic acid base recipe, one-step reaction.* Dimmer (Dimmer, 1999, BatchPlus(TM), 2002) presented a reaction scheme consisting of one reaction step (see Figure 4-6) with 93% yield at a temperature of 90° C.

	Salicylic acid	Acetic anhydride	Acetylsalicylic acid	Acetic acid
Molecular weight [g/mol]:	138	102	180	60

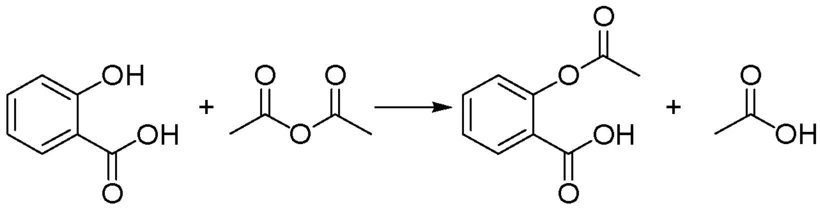


Figure 4-6: Aspirin (Acetylsalicylic acid) synthesis from Salicylic acid.

The raw materials are Salicylic acid and Acetic anhydride, while Acetic acid is used as solvent, and is also a by-product. Note that the process considered in this section is different from the investigated in Chapter 3 *Superequipment application*.

4. 3.1 Initial recipe

The recipe is projected for 10 kmol quantity of reactants. The following steps are considered:

1. Charge 1380 kg Salicylic Acid, 1428 kg Acetic Anhydride, 600 kg Acetic Acid

2. React via Aspirin synthesis (see Figure 4-6), this reaction is exotermic with a temperature ramp from 20°C to 90°C and the reaction time is set as uncertain with triangular random probability density: tripdf(360, 450, 520).

3. Crystallize Aspirin, the crystallization time is also set as uncertain, with a random probability density trirnd(200, 210, 270).

4. Centrifuge the Aspirin crystals, the centrifugation will be carried out in multiple drops, the centrifuge can process ca. 0.8 m^3 in one drop. The total time is set as uncertain with a lognormal random probability density lognrnd(4.787, 0.37).

5. Crystals are dried and are the final product (1674 kg), the drying operation duration is uncertain with a range: lognrnd(5.298, 0.2).

Table 4-4 provides additional details about the recipe.

Table 4-4: Aspirin case study with uncertainty. The uncertain variables are operation times for the operations are defined as a triangular or lognormal random probability distribution. The batch size of the projected recipe is 1674 kg (10 kmol total quantity of reactants). The ID's are explained in the Figure 2-2.

ID	Operation description	Time [min]	Volume [m³]	Operation class ID	Flags	Tempe-rature [°C]	Pressure [bar]	Lining material	Previous operation	Next operation
1	1 Charge	35	4.1	4	0	20	1.0	-1	0	2
2	2 React	trirnd(360,450,520)	4.1	32	0	90	1.0	1	1	3
3	3 Crystallize	trirnd(200,210,270)	4.1	15	2	90	1.0	-1	2	4
4	4 Centrifuge	lognrnd(4.787,0.37)	3.5	47	2	30	1.0	-1	3	5
5	5 Dry	lognrnd(5.298,0.20)	1.7	18	3	90	1.0	-1	4	–

For comparison and testing purposes a base recipe without uncertain operation times is given (Table 4-5), which is actually a subset of the recipe from Table 4-4.

Table 4-5: Aspirin case study with uncertainty, a deterministic recipe for Aspirin production. No uncertain time is defined. The batch size of the projected recipe is 1674 kg (10 kmol total quantity of reactants).

ID	Operation description	Time [min]	Volume [m³]	Operation class ID	Flags	Tempe-rature [°C]	Pressure [bar]	Lining material	Previous operation	Next operation
1	1 Charge	35	4.1	4	0	20	1.0	-1	0	2
2	2 React	450	4.1	32	0	90	1.0	1	1	3
3	3 Crystallize	210	4.1	15	2	90	1.0	-1	2	4
4	4 Centrifuge	120	3.5	47	2	30	1.0	-1	3	5
5	5 Dry	200	1.7	18	3	90	1.0	-1	4	–

Instead of defining uncertain operation times, only the most probable states are given. That means, for instance, that the expected value of the operation 2 React time (trirnd(360, 450, 520)) is 450 min. Similary the crystallization, centrifugation and drying time was determined. This recipe serves only for comparison of the stochastic vs. deterministic optimization.

4. 3.2 Plant line

The multipurpose batch plant considered in this case study (see Table 4-6) contains 6 reactors of different sizes, multidrop-centrifuges, extractors and a dryer. Additional supporting equipment, such as pumps, heat-exchangers, mills, etc., is not listed. Each equipment unit is characterized by its class, class number, nominal volume, floor (eventually coordinates in the building), lining material and TP range. The same plant is used for the deterministic Aspirin recipe comparison.

Table 4-6: Plant line used in the Aspirin uncertainty case study. The ID's are explained in the Figure 2-2.

ID	Equipment name	Equipment class name	Equipment class ID	Nominal volume [m^3]	Floor	Lining material	TP range
1	Ce0.4a	Centrifuge	135	0.4	0	5	1
2	Ce0.4b	Centrifuge	135	0.4	0	5	1
3	Cr10a2	Crystallizer	5	10	1	5	1
4	Cr10b4	Crystallizer	5	10	1	5	1
5	Cr10c4	Crystallizer	5	10	1	5	1
6	Dryer1	Dryer	16	6.3	1	5	1
7	Extract1	Extractor	6	4	1	1	1
8	Extract2	Extractor	6	4	1	5	1
9	R4a	Reactor	14	4	3	5	1
10	R4b	Reactor	14	4	3	5	1
11	R4d	Reactor	14	4	2	1	1
12	R4e	Reactor	14	4	2	5	1
13	R6a	Reactor	14	6.3	2	5	1
14	R6f1	Reactor	14	6.3	1	5	1
15	T10a	Tank	19	10	0	1	1
16	T16a	Tank	19	16	0	1	1

4. 3.3 Optimization settings

A Tabu-search method of optimization with uncertainty was used for this case study. Prioritized list of objective functions was: 1. productivity, 2. robustness. All three robustness objectives were examined in three separate runs. Robustness #3 was computed according to Eq. 32 with parameters: $c_+=0.36$; $c_-=0.36$ (see Section 2. 3.4).

For data sampling, a Latin Hypercube method resulting in 400 points was used, 400 points per each uncertain variable so, that each variable is tested in its whole range. As mentioned in the Section 2. 3 *Stochastic batch design problem*, the Latin Hypercube sampling method ensures generation of probabilistic states in the whole range of the uncertain variable values, in this case the ranges for *2 React, time* <*360, 520*> *min, 3 Crystallize, time* <*200, 270*> *min* and so forth (see Table 4-4). That means that the deterministic case study recipe (Table 4-5) is a subset of the 400 probabilistic states. In fact, the expected values of uncertain variables are stored for each stochastic recipe, in order to compute the robustness objective function.

Additionally for monovariate sensitivity analysis another 48 data sampling point combinations were defined, 12 for each variable. The random numbers were generated according to standard probability distribution functions (Equations 33–35). The data sampling procedure is repeated for each evaluated neighbour from the solution neighbourhood before the actual evaluation of productivity takes place.

For the deterministic recipe definition, the prioritized objective function list was set as: 1. productivity, 2. number of equipment. The robustness objective function can be computed only for the uncertain case study. Other settings are identical to

the stochastic TS optimization.
Other TS parameters were set equally as defined in Section 4. 2.3 *Optimization settings*.

4. 3.4 Aspirin case study results

Figure 4-7 shows the resulting Designs #3, #2 and #1. The Design #3 is identified as a non-dominated solution with objective values in expected productivity 638 kg/hr and Robustness R3 = 83%.

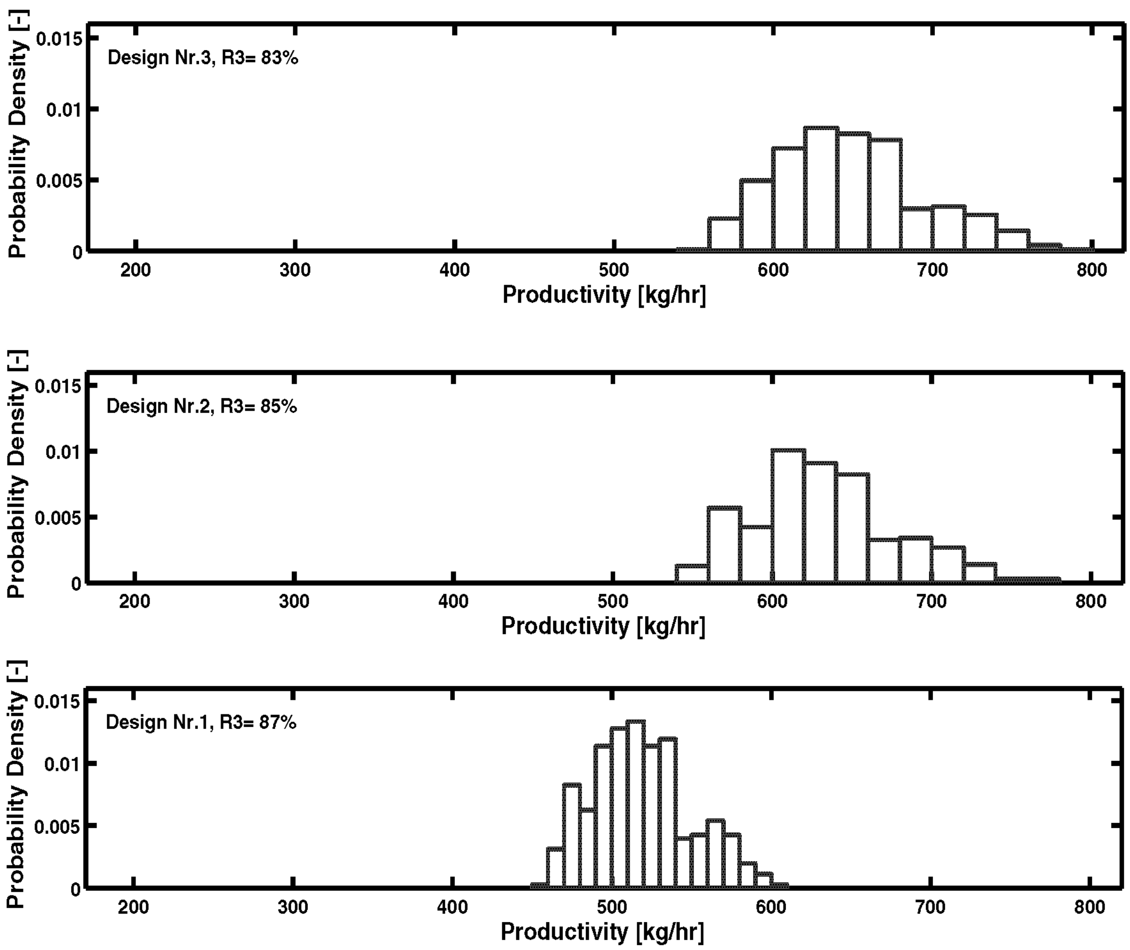


Figure 4-7: Three resulting designs from Aspirin uncertainty case study. Designs #1, #2 and #3 are displayed in a histogram of probability density vs. productivity [kg/hr] objective function obtained by Latin-Hypercube sampling method. Robustness #3 value is listed in the header of each graph.

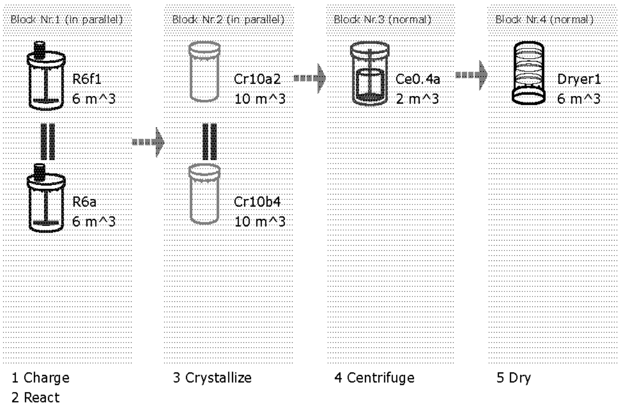
The complete list of the discussed designs' objective values is displayed in Table 4-7. The table shows that designs vary substantially in performance values, because the number of equipment units is critical here for the batch size.

Table 4-7: Results list for Aspirin uncertainty case study. The expected productivity refers to the productivity value with the highest probability density value.

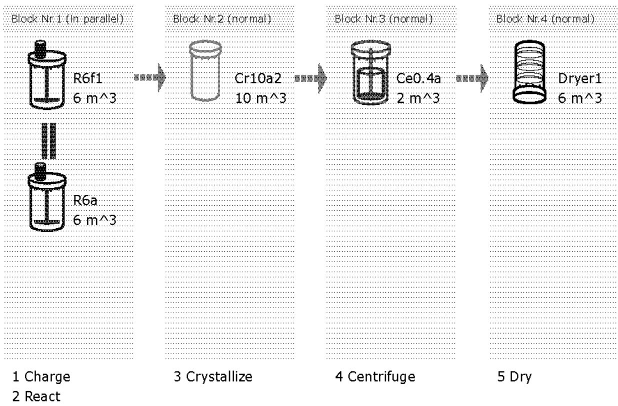
Design #	Expected productivity [kg/hr]	Robustness R1 [%]	Robustness R2 [%]	Robustness R3 [%]	Number of equipment units [pcs.]
3	638	1.0	54.4	83.0	6
2	622	1.0	57.1	84.7	5
1	510	1.4	58.2	87.0	6

Comparison of the Design #3 and #2 flowsheets (see Figure 4-8) indicates, that the time-bottleneck recipe block is the Block Nr.1 (operations 1 Charge 2 React). Indeed, the remaining operations are utilized by the same equipment units, that means the productivity value difference between the two designs is resulting from the crystallization operation.

(a) Design (#3) flowchart.
Production Rate [kg/h]=638
Number of units used [-]=6



(b) Design (#2) flowchart.
Production Rate [kg/h]=622
Number of units used [-]=5



(c) Design (#1) flowchart.
Production Rate [kg/h]=510
Number of units used [-]=6

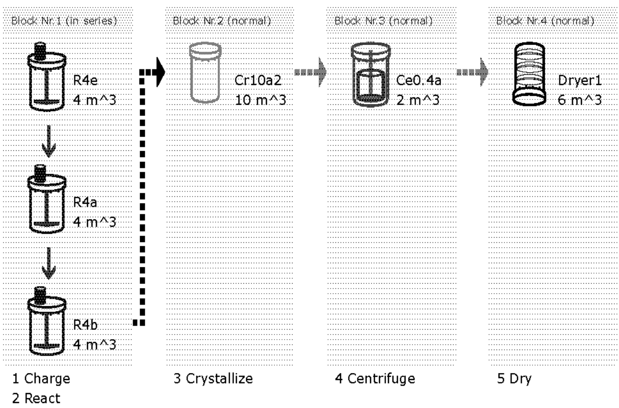


Figure 4-8: Aspirin case study with uncertainty, (a) Design #3 flowchart (b) Design #2 flowchart (c) Design #1 flowchart.

The comparison of Gantt-charts for Design #3 (Figure 4-9a), Design #2 (Figure 4-9b) and Design #1 (Figure 4-9c) shows the expected operation times and schedule for equipment units. The base case scenario is equivalent to the deterministic recipe optimization. The Design #3 has the highest productivity due to the parallel utilization of two reactors. That means that the volume bottleneck was eliminated, reactors are fully charged, but the cycle time (480 min in the expected state) is relatively high. Design #2 has shorter cycle time (390 min in the expected state), because the crystallization is not in parallel mode and thus the two reactors are not fully charged. The Design #1 utilizes three reactors in series, which lowers the cycle time to 200 minutes in the expected state. As a trade-off, a high batch size value (as of Design #1) cannot be reached, because there are only $4m^3$ reactors available.

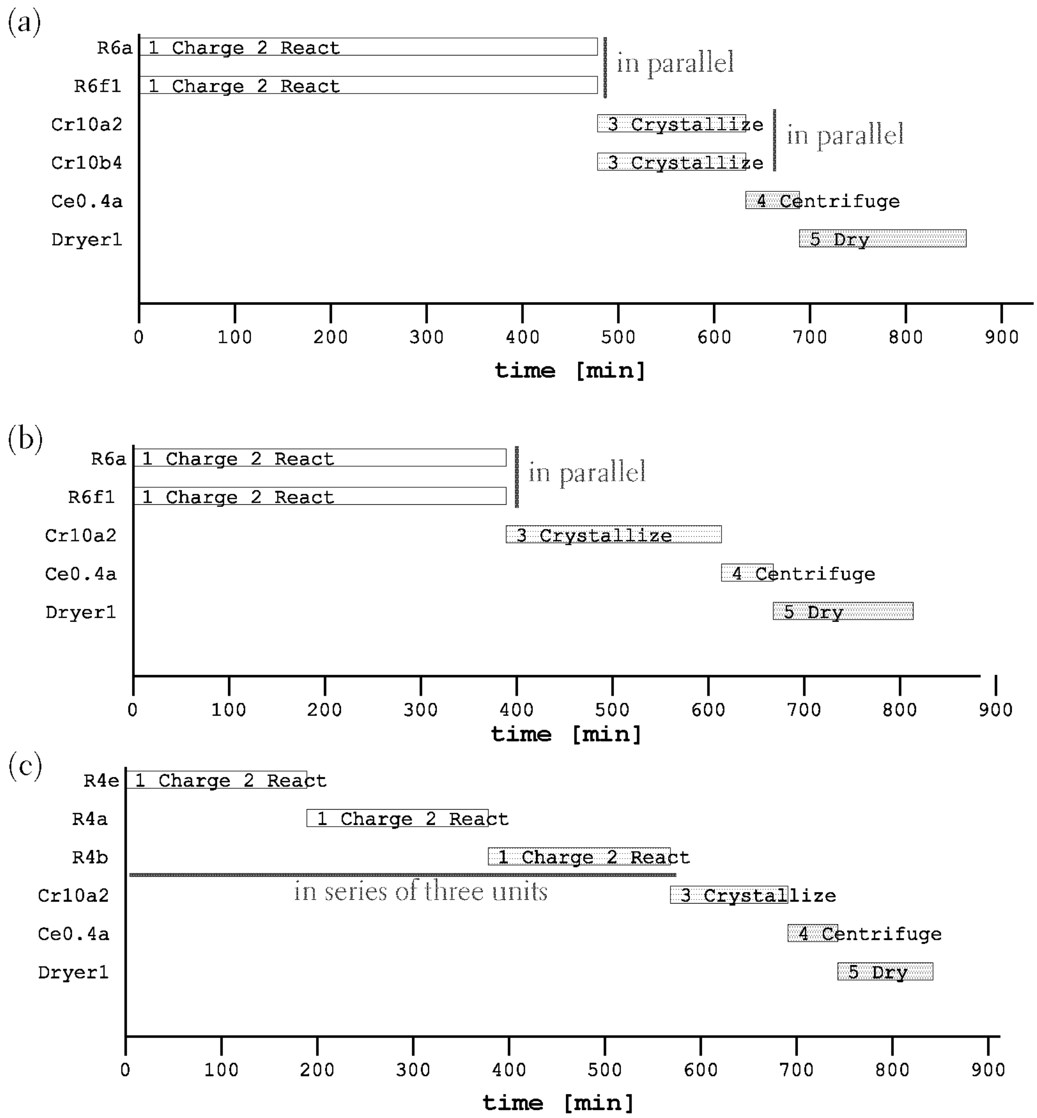


Figure 4-9: Aspirin case study with uncertainty, the Gantt-charts of (a) Design #3 (b) Design #2 (c) Design #1

The monovariate sensitivity analysis of productivity dependency from uncertain operation times is displayed on Figure 4-10. It shows the correlation of reaction time and the productivity. Other parameters are uncorrelated in the monovariate sensitivity graphs.

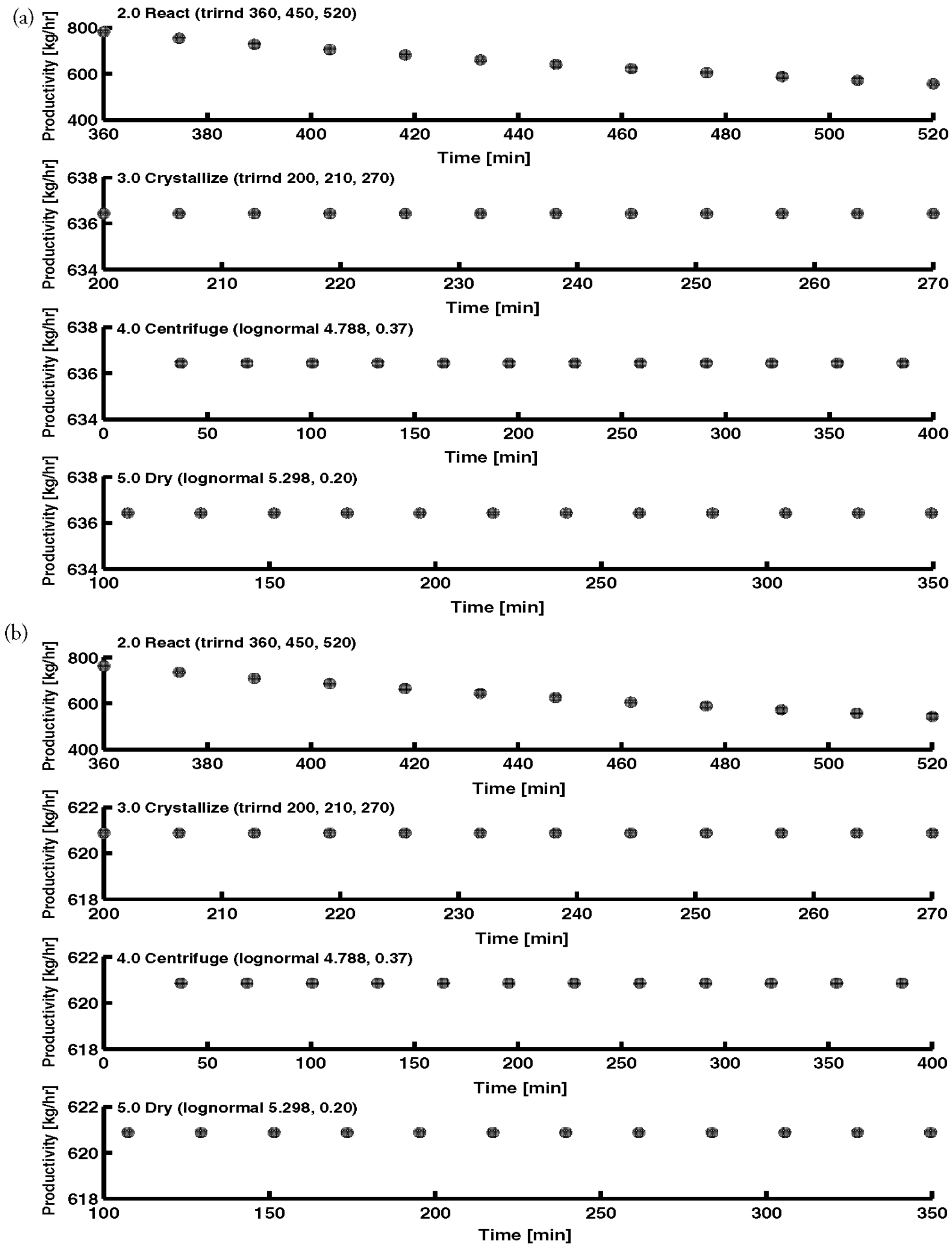


Figure 4-10: Aspirin uncertainty case study, monovariate sensitivity analysis for (a) Design #3 (b) Design #2.

Analysis of the uncertain operation time (see Figure 4-11) shows a high corellation of the reaction time (2 React) and productivity for both Designs #3 and #2. Unless the reaction time for Design #3 (Figure 4-11a) is kept under 400 minutes, the reaction will be the time-limiting operation. The same trend can be observed in Design #2 (Figure 4-11b). The operation '2 React' is also a volume limiting operation for both designs. Other operations are uncorrelated to the productivity in most of the probabilistic states.

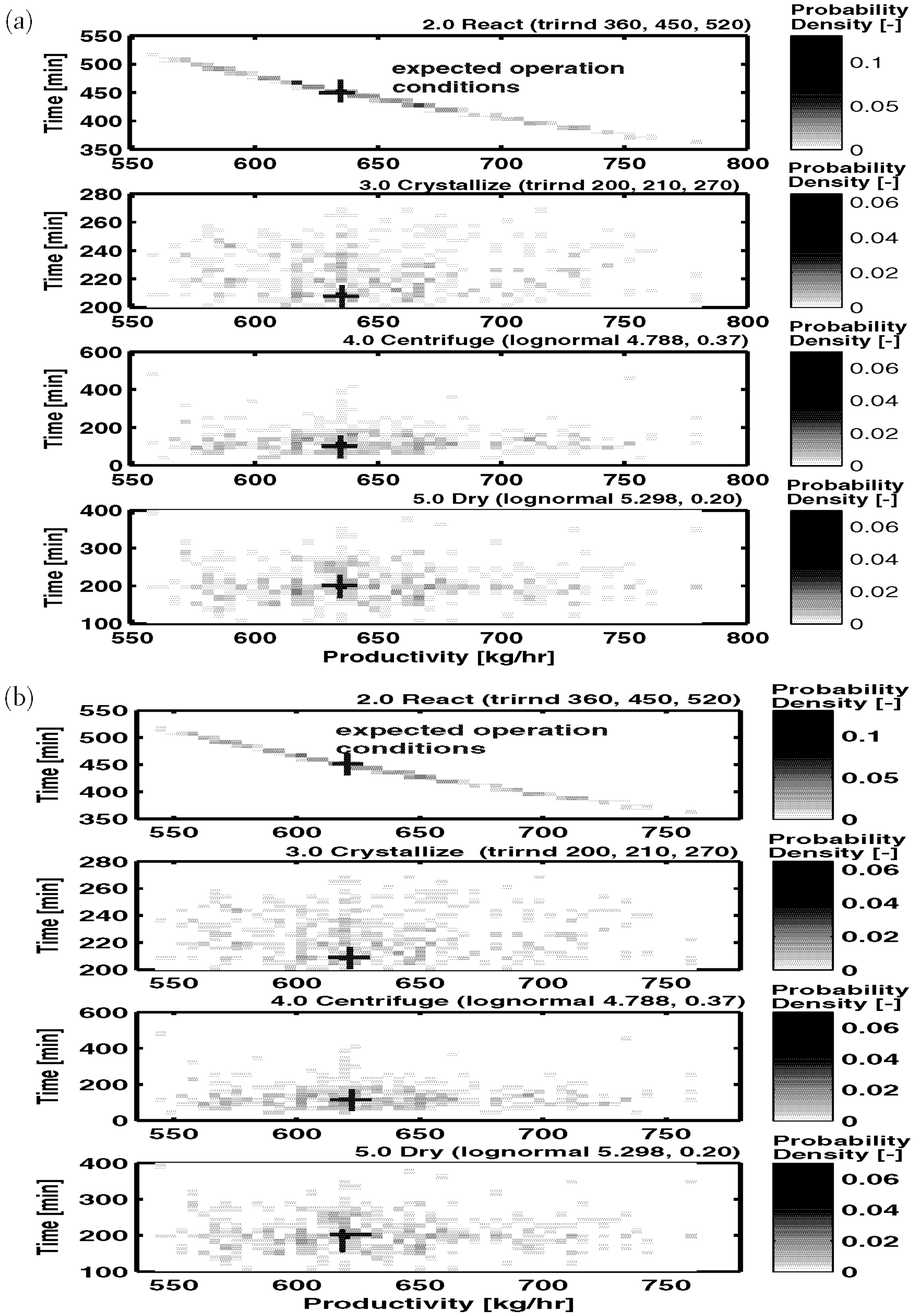


Figure 4-11: Aspirin case study with uncertainty. Multivariate sensitivity analysis with 400 Latin-Hypercube sampling points. Operation times for 2 React, 3 Crystallize, 4 Centrifuge and 5 Dry are displayed for (a) Design #3 (b) Design #2. The white-to-black scale bar represents the probability density values. The cross marks represent the state with the expected productivity and expected operation time.

The deterministic recipe TS optimization required only about 2 % of the computational time compared to the probabilistic optimization case study. Table 4-8 shows the results list.

Table 4-8: Results list for the deterministic optimization in the Aspirin case study with uncertainty.

Design	Productivity	Number of equipment units
#	[kg/hr]	[pcs.]
1	638	6
...	...	...
7	622	5
8	600	6
10	510	6
11	203	4

It is noteworthy, that this case study is relatively small and therefore the probability of identifying the non-dominated solution is high. The first design from the deterministic optimization is displayed on Figure 4-12.

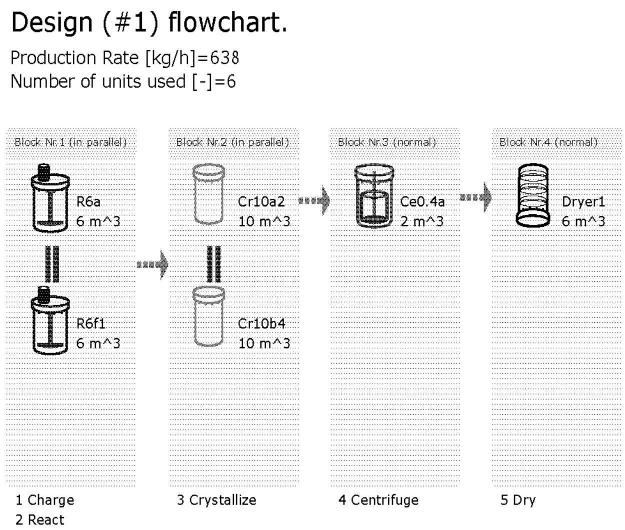


Figure 4-12: Aspirin case study, deterministic optimization. Design #1 flowsheet.

It is identical in performance and design layout to the Design #3 from the stochastic optimization case (Figure 4-8a).

4. 4 Discussion and conclusions

If we want to compare the three robustness definitions presented in this text, we might generalize, that the robustness #1 is helpful for determining the expectation of the most probable productivity, which does not provide any information about the importance of considered uncertain variables, nor the dangers of unexpected productivity drops in certain cases for the resulting designs.

Robustness #2 delivers information about the probability of achieving an expected design productivity or higher. It is a more suitable measure as an objective function than the Robustness #1, but does not take into account the intervals lower than expected productivity value.

Robustness #3 is the most versatile definition aiming at determining the most probable interval of operation of a design. If, for instance, the interval is +-25% around the probability of the expected productivity, the 50 % wide span ensures capturing of most problems related to uncertain operation of the examined solution. The presented productivity robustness method can be used as a tool for: decision making towards design implementation into the production phase, or aimed at determining which input parameters are the most important if the process is going to be implemented in a given production plant. From the point of view that a multipurpose batch plant is built for many processes as versatile as possible, there will always be a limit in equipment which prevents from further increasing the process performance. Therefore optimization in the recipe/process research has to be performed. We showed that such optimization is strongly design dependent, therefore the largest opportunities in process optimization reside in the combined optimization of equipment utilization plus process/recipe development in the early stages.

The algorithm allows also other uncertain parameters to be defined (such as varying temperature or pressure in the operation, unknown volume, etc.) which is not demonstrated in the presented case studies.

Scalability of the method is assured per se from small problems to huge sets maintaining a guaranteed feasibility and at least near-optimal quality of results.

Automatic scaling of the recipe batch size according to free capacity of the design equipment plus optimization based on the cycle time and volume utilization of a design is attained identically as by the deterministic TS algorithm. Subsequently a black-box evaluation of objective functions follows.

In the demonstrated uncertainty formulation it is not important if the operating conditions, such as filling volume, operation time, operating temperature, are planned in the R&D stage or occurring in the production phase. In both cases the uncertain variables can be seen as stressors to a static system – to a design.

From the comparison of Figures 4-5 and 4-11 we see that already the 100 (respectively 400) sampling intervals are sufficient to analyze the correlations among the operations in a design.

Comparison of the uncertain TS optimization method with the deterministic optimization shows, that the same results can be obtained, if the prioritized list of objective functions includes productivity in the first place.

In contrast to mathematical programming methods, (for example, (Rooney and Biegler, 2003)) the presented two-stage here and now stochastic programming method implemented in TS metaheuristic algorithm offers many advantages:

- The result is not a single design, but rather a set of various good-performing designs. This advantage is inherited from the TS optimization method.
- The problem formulation does not have to be linearized or simplified in order to obtain feasible results in a convenient time-frame. The proposed solving method can handle continuous, discrete, smooth or non-smooth functions included in the mathematical models. Because of the effective neighborhood random search strategy (especially the TS move definitions), the algorithm benefits in the computational speed.
- Black-box objective functions assure general problem formulation valid for any prioritized combination of objective functions. The objective functions are independent of the TS optimization algorithm. Therefore the searching algorithm is manipulating the solutions, i.e. the designs, which are then evaluated. Sorting and decisions according to the prioritized list of objective functions then follows.
- Feasibility regions or constraint violations are not critical for the solutions, because the TS strategy assures valid solutions to be generated throughout the whole optimization process if the problem is solvable.
- The type of input variable PDF does not influence the optimization algorithm. The PDF can be defined as discrete or continuous, smooth or non-smooth function and the black-box evaluation of objective functions independent of the TS optimization core assure convergence stability, which could be a problem with non black-box optimization methods.

Conclusions and further research perspectives

5

Abstract

Due to the high degree of automation, the methodology presented in this thesis efficiently supports the design of batch processes for a single recipe. The techniques help to determine promising design alternatives in the preliminary design phase.

The superequipment method shows good applicability on a range of discussed problems. Superequipment can be equivalently extended to similar problem settings.

The stochastic problem definition included only a subset of possible probabilistic variable definitions in the specifications. A new objective function type includes the probabilistic definitions of productivity and performance robustness, which are handled by discrete approximation – a Latin Hypercube method. The presented stochastic optimization technique successfully demonstrated the potential of multiobjective algorithms involving uncertain parameters.

We therefore consider the present work as a first step in dealing with nonlinear stochastic optimization in batch design problems, suggesting many possible future research directions, rather than a completed exploration of the subject.

5. 1 Conclusions

The research work described in this thesis is concerned with the study of preliminary batch process design, more particularly with the design of methods and algorithms for solving problems related to identifying a set of performing batch designs for a given chemical recipe.
In the presented text an automated methodology in form of a set of algorithms for solving the following scope of problems was successfully introduced, investigated and applied to case studies:

1. Deterministic nonlinear multiobjective optimization of batch process designs.
2. Equipment related problems such as: grass root design of a new plant, retrofit of an old multipurpose batch plant, plant line selection out of multiple choices.
3. Stochastic multiobjective batch design optimization for a single recipe.

All the investigated problems are applicable to multipurpose batch plants.

Deterministic multiobjective-optimal set of performing process designs

First the mathematical formulation of a batch design together with methods and algorithms of: manipulating the design, computing performance characteristics and postprocessing the designs were defined in Chapter 2.

The batch design was defined as a sequence of allocation of chemical recipe operations to equipment units. Often a small number of equipment units combined with a small number of recipe tasks results in a nonlinear allocation problem of exponential size. Therefore an efficient metaheuristic optimum solving algorithm – the Tabu Search was customized and applied to this problem set.
As a result of this customizing, the modified TS can, in a short time period effectively approach the preliminary batch design problems. In the deterministic definition of the batch design problem, the solutions consist of a list of dominating (approximated Pareto-optimal) designs, local optimum designs and several good-performing designs. This list is the main output result for the decision maker, usually a process engineer in a batch plant.

The method can efficiently solve the deterministic batch process design problem. Of course, the presented results are based on assumptions and mathematical models, which cannot include the complete set of decision variables. Therefore the algorithms and software are ment as a complementary set of tools for the decision making engineers. The discussions and testing with industry experts showed the applicability, a sufficient precision of the computed results for the preliminary design phase and a good match of the blind-test results with the designs already implemented in the production.

Superequipment - batch equipment related problems

As stated in the Chapter 1, new methods for solving efficiently, the preliminary batch process design problem are needed. As a result, a novel method for efficiently reducing the combinatorial domain size was introduced (see Chapter 3). Instead of defining equipment units in the combinatorial way[8], a mathematical model called superequipment including all the properties was proposed. The major implication of this research is that instead of optimizing a large number of explicit equipment units, it is only necessary to define one superequipment unit, or a set of superequipment units in a TS optimization.

Practical applications include all problems related to adding or exchanging units in the plant. In the case of plant retrofit, the aim is to find the best possible equipment unit according to cost criteria. In the combination with NPV objective function the problem of identifying the most effective units combination as an investment can be successfully solved by proposed algorithm. Results include a set of optimal designs for a given recipe. If the investment is feasible, that means the NPV > 0, the results will automatically list the best investment scenarios at the top positions. The top selected designs contain superequipment units transformed into real equipment units which correspond to the investment propositions.

If the task is to select a plant line out of several optional lines, the optimization problem can be processed several times, or, as proposed in this thesis, only one optimization run is sufficient to identify the appropriate plant line. Solely the superequipment units are used for generating the solutions. During each iteration a design consisting only of superequipment units is transformed into a real design alternative. After this transformation the comparison and scaling of designs equipment according to the units present in the existing plant lines take place.
The results set contains in this case a sorted list of designs according to the defined objective functions and information about the plant line in which the design will be performed. Thus the decision maker can immediately see which performance can be achieved by a given plant line. This information is always design specific.

Another problem set involves a grass root design of a new multipurpose batch plant. A new method involving a specified number of superequipment units which will again result in a set of dominating (approximated Pareto-optimal) designs was shown.

One of the major conclusions is that this optimization method significantly reduces the design domain space and thus reducing the number of combinations of possible equipment units in a design. As a result, the approximated Pareto-optima can be found in less time with higher chances of finding the global optimum. Although the superequipment transformation requires additionally computational time, overall the computation is faster due to the combinatorial domain reduction.

The superequipment concept shows on practical examples that a simple mathematical model of a unit together with design heuristics can deliver superior result sets faster as compared to previously used standard Tabu Search optimization algorithms using explicit equipment unit definition.

8. by including information about equipment class, size, lining material, maximal pressure, temperature, etc.

Stochastic multiobjective batch design optimization problem

The objective of this research was to introduce uncertain variables into the batch design problem. Mathematical formulation of a stochastic batch design optimization problem taking into account uncertain recipe variables was stipulated as a solution to this problem with a practical application on a case study.

In **Chapter 4** a demonstration of uncertain recipe variables such as: operation duration (or possibly operation volume) in a multiobjective optimization shows several important conclusions related to the preliminary stage process design:

- a simple sensitivity analysis of uncertain variable to the design‘s performance is not sufficient for determining the synergic effects of several uncertain factors to productivity
- it is critical to consider and determine not only the uncertain variable by itself, but the uncertain variable set influencing the final design‘s layout and vice versa. That means, that measuring and narrowing the probabilistic distributions of such variables has to be always considered in the connection with a specific design, because in one design layout the given variable can play a crucial role, e.g. is time or volume limiting, and in another design it can become time or volume limiting only after some other conditions are met

A new stochastic objective function was defined which aims at quantifying the effect of uncertain time or volume operation variables on productivity. The robustness objective, if used in a ranked multiobjective optimization, can deliver good-performance and, at the same time, stable design sets. That means that the variance induced by stochastic recipe variables on productivity is minimized.

From a practical point of view, the deterministic optimization results usually in peak-productivity optimal designs, whereas the proposed robustness objective algorithm ensures that the productivity of a design will be in X % of the probabilistic states at least as good as expected, or that the productivity will stay within the desired interval around the expected value. The percentage of the robustness parameters can be specified at the input.

5. 2 Further research perspectives

The themes presented in this thesis suggest a new range of research topics. Some of the most promising research problems with regard to easy aggregation with the existing models and algorithms are listed below.

Ecology and energy assessment

Ecological evaluation could be implemented to integrate for instance the energy consumption predictions in the process design.

The goal of such minimization would be to identify the best design set according to prioritized objective functions: 1. productivity/costs 2. ecologic objective function. Using this objective function would also suggest additional information for the plant selection problem.
Energy consumption prediction is partially implemented by the algorithms presented in this thesis in the economic objective function, where the costs of energy and utilities are minimized.

Algorithmic advancements

The modular TS algorithm allows definition of a number of problems to be solved in one course. This opens new batch design problem solving opportunities as discussed below.

A number of recipes (N) to be computed simultaneously in one plant

In this type of optimization, a recipe matrix of a size [$m \times n$] (rows, columns) is expanded to a size [$m \times n \times N$], that means that the evaluation function will output for example not one productivity value, but N productivity values for the N specified recipes. At the present stage, this is implemented in the uncertainty computations.
In the future, it might be easily modified to accept recipes of various sizes, e.g. [$m1 \times n1$], [$m2 \times n2$], ... which would allow to test the whole portfolio of recipes and propose the best design or the suitable set of equipment units for a given portfolio. For example, in order to gain factory building space for new equipment, it is necessary to eliminate some of the old units. This optimization would propose a buying list of suitable units for the given chemical recipe portfolio.

Number of plants to be optimized simultaneously for N recipes.

This is partially solved by a classical multiple lines selection in more optimization runs, or by a superequipment method in one optimization run (see **Chapter 3**). These methods were programmed only for one recipe to be handled by the batch plants.
Accommodation of the modular TS algorithm provides additionally means for exchanging the information about the plant lines during the optimization. This means, that the plant lines could be optimized in a cooperative way, as is usual in equipment pool plants. The problem definition then changes to maximize profit of a portfolio of chemical recipes in a number of plant lines over a period of time. The plant lines could be defined as totally independent, i.e. a multinational company producing in different countries, or as a totally cooperative plant (Equipment pool) or a state between these two limiting cases.

Parallelization

The problems can be easily parallelized. For successful parallelization the modular TS algorithm offers: vectorized evaluation of objective functions, multidimensional recipes, solution neighbourhood which can be split into two or more parts, move definitions which can be distributed to slave machines according to cpu time needed for such move application on a given set of solution candidates.

Tabu Search advanced options

We utilize simple tabu tenure, aspiration, stopping criteria, etc. as the TS options. Long term memory could be used for intensifying the search in promising solution areas. Interconnected tabu tenure lists for the cases where multiple problems are being solved simultaneously could improve the convergence speed. All prerequisites are met in the advanced TS algorithm presented in this thesis.

Recipe optimization

In the current stage, the recipe remains fixed in the material balance. That means that the resulting design‘s batch size is scaled according to the precomputed mass balance. Operation times are scaled according to simple rules. Mass balance computation model and additional simple models for separation steps or reaction kinetics would be beneficial in connection with the current optimization tool and with the robustness definition. This would enable the decision maker to see directly the effects of: solvent selection, separation step selection or reaction kinetics on a production stage design.

As we concluded in the Chapter 4, recipe parameters importance is strongly dependent on the design or plant line of implementation. Therefore study of reaction yield itself might not bring the desired benefits in design's productivity, whereas in connection with the existing plant and design optimization techniques the importance of a reaction yield for the examined operation can be easily determined.

5. 3 Final conclusions

Due to the high degree of automation, the methodology efficiently supports the design of batch processes for a single recipe. The methodology helps to avoid overlooking particular problems, or particularly promising design alternatives in the preliminary design phase.

The superequipment method shows good applicability on a range of problems, from which only a few were discussed.

The stochastic problem definition included only a subset of possible probabilistic variable definitions in the specifications. A new objective function type: the probabilistic definitions of productivity and performance robustness, which are handled by discrete approximation – a Latin Hypercube method – successfully demonstrated the potential of multiobjective algorithms involving uncertain parameters.

We therefore consider the present work as a considerable first step in dealing with nonlinear stochastic optimization in batch design problems, suggesting many possible future research directions, rather than a completed exploration of the subject.

List of symbols and abbreviations

6. 1 List of Symbols

A	assignment matrix, lists operation classes allowed in each equipment class
b	a vector contained in *B* linking the recipe indexes R.ID to a block k_i
B	block recipe matrix, each block k_i of *B* contains one or more recipe steps
BS	batch size of a design [t/batch]
$c_{1,2}$	term denoting the constraints related to uncertain optimization problem definition used in the first (1) and second (2) stage
$c_{+,-}$	predefined bias in the CDF of design's productivity G
cdf()	cumulative distribution function
C	a constant or a parameter used in the PDF, CDF functions
CT	cycle time of a batch design [min]
E	equipment list, matrix, list of all equipment in the given plant
E(ξ)	expectation value of ξ in the stochastic problem definition
E_{buy}	list of equipment that can be added to an existing plant, matrix
EqClass	equipment class, used in *A*, *E*, *L* matrices
f(x)	function of x
$f_{1,2}(x)$	the first/second stage deterministic objective function
G	productivity in the stochastic optimization definition [kg/hr]
H(B, u)	heuristics as a function of block recipe matrix and user-selected variables u
ID	identification number, used in connection with matrices B, R, A, E, EqClass, OpClass, TP
k_i	single block from block recipe matrix *B*
L	single design, ordered sequence of assignments of given recipe blocks k_i into equipment units E_j
Lining	lining material

lognrnd(log(μ), σ) random lognormal probability distribution function; returns an array of random numbers chosen from a lognormal distribution with mean μ and standard deviation σ

m constraints in the stochastic function definition, m_1 refer to the first stage, m_2 to the second stage of the formulation

$\overline{m}$ constraints in the stochastic function definition, see the definition of *constraint m*

M move set, matrix containing the set of possible moves in tabu search optimization

mode() function returning a value from a list of input arguments which has the maximum value in the pdf() function

N a number of elements in a vector

N_{eq} number of equipment units [pcs.]

normrnd(μ, σ) random normal probability distribution function; returns an array of random numbers chosen from a normal distribution with mean μ and standard deviation σ

NPV Net Present Value [USD, CHF]

oc operation class equivalent to OpClass, used in the Superequipment definitions

OpClass operation class used in *E* and *A* matrix

P pressure, in TP range of equipment: the minimal/maximal operating pressure range for the apparatus [bar], P is also used as a symbol for plant matrix definition.

pdf() probability distribution function

Q(x) the second stage objective function including uncertain parameters

r recipe step, one row of matrix *R*

R recipe, matrix, recipe consists of process steps
R.Duration - task duration, R.LiningID - lining material ID, R.OpClassID operation class ID, R.Pressure - maximal pressure for operation, R.Temperature - maximal temperature for operation

R, $R_{1,2,3}$ robustness of a design in a stochastic optimization, the subscripts refer to the first, second or third robustness definition

S superequipment class

t constraint in the stochastic optimization problem

T temperature, in TP range of equipment: the minimal/maximal operating temperature range for the apparatus [°C], Tabu List definition matrix

TP temperature and pressure range of equipment, matrix [°C], [bar]

trirnd() triangular probability density function

u user defined variables in the heuristics

U	feasible equipment classes matrix
V	eligible equipment units for given recipe block k_i
W	design candidate list resulting from the application of the moves *M* to the actual design L
x	first-stage decision variables in the stochastic function definition
X	operations in the recipe block k_i
X_i	input variables Xi, used in the stochastic function definition
y	second stage variables in the stochastic optimization problem
Y	the most probable value
z(x)	stochastic objective function
ω	random event from a set of random events Ω
Ω	set of random events
ξ	uncertain variables, random vector

6. 2 List of indexes

°	current item, for instance in *L*, *M*.
add	addition, in move definition *M*
eq	equipment units, in number of equipment units N
G	productivity related index, used in connection with pdf() and cdf()
i, *j*	counters over operations or equipment or elements of the recipe such as blocks or steps
m	index of *M*, by extension, index on *L*. *L* is the design resulting from the move *M*
max	maximal, for instance in TP range
min	minimal, for instance in TP range
p	in parallel, in move definition M
rem	removal, in move definition *M*
repl	replace, in move definition *M*
s	in series, in move definition M
ξ	index refering to function of uncertain variables
+,-	value higher resp. lower than the given value

6. 3 List of abbreviations

CDF	Cumulative Distribution Function
EA	Evolutionary Algorithm
Eq	Equipment
EPP	Equipment Pool Plant
GA	Genetic algorithm
HTML	Hypertext Markup Language
ID	Identification number
MI(N(LP))	Mixed Integer (Non(linear Programming))
MO	Multi-objective
MOGA	Multi Objective Genetic Algorithms
NP	non-deterministic polynomial time
PDF	Probability Density Function
R	Reactor
R&D	Research and development
SA	Simulated annealing
SEq	Superequipment unit, used in diagrams
SuperEq	Superequipment unit
TS	Tabu Search
XML	eXtended Markup Language

References

[Ahmed and Sahinidis, 1998]Ahmed, S. and Sahinidis, N. V. (1998). Robust process planning under uncertainty. *Industrial & Engineering Chemistry Research*, 37:1883–1892.

[Allgor et al., 1996]Allgor, R. J., Barrera, M. D., Barton, P. I., and Evans, L. B. (1996). Optimal batch process development. *Computers & Chemical Engineering*, 20(6–7):885–896.

[Allgor et al., 1999]Allgor, R. J., Evans, L. B., and Barton, P. I. (1999). Screening models for batch process development part 1. design targets for reaction/distillation networks. *Chemical Engineering Science*, 54:4145–4164.

[Areibi et al., 2001]Areibi, S., Moussa, M., and Abdullah, H. (2001). A comparison of genetic/memetic algorithms and heuristic searching. In *International Conference on Artificial Intelligence (IC-AI), http://citeseer.ist.psu.edu/440439.html*.

[Armentano and Arroyo, 2004]Armentano, V. A. and Arroyo, J. E. C. (2004). An application of a multi-objective tabu search algorithm to a bicriteria flowshop problem. *Journal of heuristics*, 10:463–481.

[Avriel and Wilde, 1969]Avriel, M. and Wilde, D. J. (1969). Engineering design under uncertainty. *Industrial & Engineering Chemistry and Process Design Development*, 8(1):124–131.

[Balasubramanian and Grossmann, 2003b]Balasubramanian, J. and Grossmann, I. E. (2003b). Scheduling optimization under uncertainty – an alternative approach. *Computers & Chemical Engineering*, 27:469–490.

[Balasubramanian and Grossmann, 2003a]Balasubramanian, J. and Grossmann, I. E. (2003a). Scheduling optimization under uncertainty–an alternative approach. *Computers & Chemical Engineering*, 27(4):469–490.

[Bansal et al., 2002]Bansal, V., Perkins, J. D., and Pistikopoulos, E. N. (2002). Flexibility analysis and design using a parametric programming framework. *AIChE Journal*, 48(12):2851–2868.

[Barbosa-Povoa and Macchietto, 1993]Barbosa-Povoa, A. P. and Macchietto, S. (1993). Optimal design of multipurpose batch plants 1. problem formulation. *Computers & Chemical Engineering*, 17:S35–S38.

[Barbosa-Povoa and Macchietto, 1994]Barbosa-Povoa, A. P. and Macchietto, S. (1994). Detailed design of multipurpose batch plants. *Computers & Chemical Engineering*, 18(11-12):1013–1042.

[Bastin, 2004] Bastin, F. (2004). *Trust-Region Algorithms for Nonlinear Stochastic Programming and Mixed Logit Models*. PhD thesis, Facultes Universitaires Notre-Dame de la Paix Namur.

[BatchPlus(TM), 2002]BatchPlus(TM) (1998-2002). http://www.aspentech.com/brochures/batchplus.pdf. *Aspen Technologies Inc.*, www.aspentech.com.

[Bernardo et al., 2001]Bernardo, F. P., Pistikopoulos, E. N., and Saraiva, P. M. (2001). Quality costs and robustness criteria in chemical process design optimization. *Computers & Chemical Engineering*, 25(1):27–40.

[Biegler et al., 1997]Biegler, L. T., Grossmann, I. E., and Westerberg, A. W. (1997). *Systematic Methods of Chemical Process Design*. Prentice Hall, New Jersey.

[Bonfill et al., 2005]Bonfill, A., Espuna, A., and Puigjaner, L. (2005). Addressing robustness in scheduling batch processes with uncertain operation times. *Industrial & Engineering Chemistry Research*, 44:1524–1534.

[Brealey and Myers, 1996]Brealey, R. E. and Myers, S. C. (1996). *Principles of corporate finance, 5th ed.* McGraw-Hill, New York.

[Cavin, 2003] Cavin, L. (2003). A systematic approach for multi–objective process design in multi–purpose batch plants. *PhD. Thesis Nr. 14898*, ETH, Zurich.

[Cavin et al., 2004]Cavin, L., Fischer, U., Glover, F., and Hungerbühler, K. (2004). Multi-objective process design in multi-purpose batch plants using tabu search optimization algorithm. *Computers & Chemical Engineering*, 28(4):459–478.

[Cavin et al., 2005]Cavin, L., Fischer, U., Mosat, A., and Hungerbühler, K. (2005). Batch process optimization in a multipurpose plant using tabu search with a design-space diversification. *Computers & Chemical Engineering*, 29(8):1770–1786.

[Chen et al., 2003]Chen, C. L., Lin, R.-H., and Zhang, J. (2003). Genetic algorithms for MD-optimal follow-up designs. *Computers & Operations Research*, 30(2):233–252.

[Cowan, 2003] Cowan, G. (2003). Chemical engineering plant cost index from 1950 to 2003. http://members.rogers.com/fhcurry/ceirev3.xls.

[Dedieu et al., 2003]Dedieu, S., Pibouleau, L., Azzaro-Pantel, C., and Domenech, S. (2003). Design and retrofit of multiobjective batch plants via a multicriteria genetic algorithm. *Computers & Chemical Engineering*, 27:1723–1740.

[Dietz et al., 2006]Dietz, A., Azzaro-Pantel, C., Pibouleau, L., and Domenech, S. (2006). Multiobjective optimization for multiproduct batch plant design under economic and environmental considerations. *Computers & Chemical Engineering*, 30(4):599–613.

[Dimmer, 1999]Dimmer, P. (1999). *Unsicherheits- und Sensitivitätsberechnung in der frühen Kostenschätzung von Batchverfahren unter besonderer Berücksichtigung der Abfallentsorgung*. PhD thesis, Nr. 13337, ETH Zurich.

[Espuna et al., 1989]Espuna, A., Lazaro, M., Martinez, J. M., and Puigjaner, L. (1989). An efficient and simplified solution to the predesign problem of multiproduct plants. *Computers & Chemical Engineering*, 13(1/2):163–174.

[FDA, 2006] FDA (2006). Good manufacturing practice guide for active pharmaceutical ingredients. *www.fda.gov.*

[Garey and Johnson, 1979]Garey, M. and Johnson, D. (1979). *Computers and intractability: a guide to the theroy of NP-completeness.* W.H. Freeman, 1st edition.

[Glover and Laguna, 1997]Glover, F. and Laguna, M. (1997). *Tabu Search*. Dordrecht, Kluwer academic publisher groups.

[Gross and Roosen, 1998]Gross, B. and Roosen, P. (1998). Total process optimization in chemical engineering with evolutionary algorithms. *Computers & Chemical Engineering*, 22(Suppl.):S229–S236.

[Grossmann and Floudas, 1987]Grossmann, I. E. and Floudas, C. A. (1987). Active constraint strategy for flexibility analysis in chemical processes. *Computers & Chemical Engineering*, 11(6):675–693.

[Grossmann et al., 1983]Grossmann, I. E., Halemane, K. P., and Swaney, R. E. (1983). Optimization strategies for flexible chemical processes. *Computers & Chemical Engineering*, 7(4):439–462.

[Grossmann and Sargent, 1978]Grossmann, I. E. and Sargent, R. W. H. (1978). Optimum design of chemical plants with uncertain parameters. *AIChE Journal*, 24(6):1021–1028.

[Hess et al., 2004]Hess, S., Train, K. E., and Polak, J. W. (2004). On the use of a modified latin hypercube sampling (mlhs) method in the estimation of a mixed logit model for vehicle choice, submitted article. *Transportation Research*, 40:147–163.

[Hungerbühler et al., 1999]Hungerbühler, K., Ranke, J., and Mettier, T. (1999). *Chemische Produkte und Prozesse, Grundkonzepte zum umweltorientierten Design*. Springer Verlag, Berlin.

[Jankowitsch, 2000]Jankowitsch, O. (2000). Environmental assessment of waste treatment process alternatives. *Master Thesis, ETH Zurich, Switzerland.*

[Kolbe, 1860] Kolbe, H. (1860). Acetylsalicylic acid synthesis. *Ann*, pages 113–125.

[Kubic and Stein, 1988]Kubic, W. L. and Stein, F. P. (1988). A theory of design reliability using probability and fuzzy sets. *AIChE Journal*, 34(4):583–601.

[Lin et al., 2005]Lin, B., Chavali, S., Camarda, K., and Miller, D. C. (2005). Computer-aided molecular design using tabu search. *Computers & Chemical Engineering*, 29(2):337–347.

[Lin and Miller, 2004]Lin, B. and Miller, D. C. (2004). Tabu search algorithm for chemical process optimization. *Computers & Chemical Engineering*, 28(11):2287–2306.

[Lindsey and Jeskey, 1957]Lindsey, A. S. and Jeskey, H. (1957). Acetylsalicylic acid synthesis from 4-hydroxybenzoic acid. *Chem. Rev.*, 57 (4):583–620.

[Loonkar and Robinson, 1970]Loonkar, Y. and Robinson, J. D. (1970). Minimization of capital investment for batch processes. calculation of optimum equipment sizes. *Industrial & Engineering Chemistry and Process Design Development*, 9:625.

[Marchal, 2003]Marchal, B. (2003). *XML by example*. QUE, Indianapolis, Indiana.

[Matches, 2006]Matches (2006). Process equipment cost estimates. http://www.matche.com/equipcost/ from 14.5.2006.

[Mauderli and Rippin, 1979]Mauderli, A. and Rippin, D. W. T. (1979). Production planning and scheduling for multi-purpose batch chemical plants. *Computers & Chemical Engineering*, 3:199–206.

[Michalewicz, 1994]Michalewicz, Z. (1994). *Genetic Algorithms + Data Structures = Evolution Programs*, volume second, extended edition. Springer, Dept. of Computer Science, Univ. of North Carolina, Charlotte, NC 28223, USA. zbyszek@mosaic.uncc.edu.

[Mockus and Reklaitis, 1999]Mockus, L. and Reklaitis, G. V. (1999). Continuous time representation approach to batch and continuous process scheduling. 1. MINLP formulation. *Industrial & Engineering Chemistry Research*, 38:197–203.

[Mosat et al., 2004]Mosat, A., Cavin, L., Fischer, U., and Hungerbühler, K. (2004). Multipurpose batch plant optimization with cost considerations in the preliminary design phase. In *Slovak Society of Chemical Engineering, 33rd conference*.

[Mosat et al., 2005a]Mosat, A., Cavin, L., Fischer, U., and Hungerbühler, K. (2005a). Multiobjective optimization of multipurpose batch plants using superequipment class approach. *submitted to Computers & Chemical Engineering Journal*, Nov.

[Mosat et al., 2005b]Mosat, A., Cavin, L., Fischer, U., and Hungerbühler, K. (Nov. 2005b). Multiobjective optimization of multipurpose batch plants using superequipment class concept. In *AIChE Annual Meeting, Cincinnati*.

[Mulvey et al., 1995]Mulvey, J. M., Vanderbei, R. J., and Zenios, S. A. (1995). Robust optimization of large-scale systems. *Operations Research*, 43(2):264–281.

[Niedrig, 2004]Niedrig, B. (2004). Process optimization in multipurpose batch plants considering investment in additional equipment. Master's thesis, Institute for Chemical- and Bioengineering Safety and Environmental Technology Group ETH Zürich, Switzerland.

[Nishida et al., 1974]Nishida, N., Ichikawa, A., and Tazaki, E. (1974). Synthesis of optimal process systems with uncertainty. *Industrial & Engineering Chemistry and Process Design Development*, 13(3):209–214.

[NIST, 2006] NIST (05.2006). Np - completness. *http://www.nist.gov/dads/HTML/npcomplete.html*.

[Oster and Fechtel, 2002]Oster, B. and Fechtel, U. (2002). *Ullmanns Encyclopedia of Industrial Chemistry*. Wiley-VCH Verlag GmBH & Co.

[Painton and Diwekar, 1995]Painton, L. and Diwekar, U. (1995). Stochastic annealing for synthesis under uncertainty. *European Journal of Operational Research*, 83(3):489–502.

[Papageorgaki and Reklaitis, 1990]Papageorgaki, S. and Reklaitis, G. V. (1990). Optimal design of multipurpose batch plants. 1. problem formulation. *Industrial & Engineering Chemistry Research*, 29:2054–2062.

[Papageorgaki and Reklaitis, 1993]Papageorgaki, S. and Reklaitis, G. V. (1993). Retrofitting a general multipurpose batch chemical plant. *Industrial & Engineering Chemistry Research*, 32:345–362.

[Patel et al., 1991]Patel, A. N., Mah, R. S. H., and Karimi, I. A. (1991). Preliminary design of multiproduct noncontinuous plants using simulated annealing. *Computers & Chemical Engineering*, 15(7):451–469.

[Perry and Green, 1997]Perry, R. H. and Green, D. W. (1997). *Perry's chemical engineers' handbook – 7th ed.* McGraw-Hill, New York.

[Petrides et al., 2002]Petrides, D. P., Koulouris, A., and Lagonikos, P. T. (2002). The role of process simulation in pharmaceutical process development and product commercialization. *Pharmaceutical Engineering*, 22(1):1–8.

[Pham and Karaboga, 2000]Pham, D. T. and Karaboga, D. (2000). *Intelligent Optimisation Techniques*. Springer Verlag, London. ISBN 1-85233-028-7.

[Reichstein and Grüssner, 1935]Reichstein, T. and Grüssner, A. (1935). Eine ergiebige Synthese der L-Ascorbinsäure (C-Vitamin). *Helvetica Chimica Acta*, 17:311–328.

[Reklaitis, 1989]Reklaitis, G. V. (1989). Progress and issues in computer aided batch process design. *Proc. 3rd Int. Conf. Found. Comput.-Aided Process Des.*, 3:241–276.

[Reklaitis, 1995]Reklaitis, G. V. (1995). Scheduling approaches for the batch process industries. *ISA Transactions*, 34(4):349–358.

[Rippin, 1983] Rippin, D. W. T. (1983). Design and operation of multiproduct and multipurpose batch chemical plants. - an analysis of problem structure. *Computers & Chemical Engineering*, 7:463–481.

[Rippin, 1991] Rippin, D. W. T. (1991). Batch process planning. *Chemical Engineering Science*, 98:100–107.

[Rippin, 1993] Rippin, D. W. T. (1993). Batch process systems engineering: A retrospective and prospective review. *Computers & Chemical Engineering*, 17:1–13.

[Rooney and Biegler, 2003]Rooney, W. C. and Biegler, L. T. (2003). Optimal process design with model parameter uncertainty and process variability. *AIChE Journal*, 49(2):438–449.

[Sahinidis, 2004]Sahinidis, N. V. (2004). Optimization under uncertainty: state-of-the-art and opportunities. *Computers & Chemical Engineering*, 28(6-7):971–983.

[Saliby and Pacheco, 2002]Saliby, E. and Pacheco, F. (2002). An empirical evaluation of sampling methods in risk analysis simulation: quasi monte carlo, descriptive sampling, and latin hypercube sampling. In Yucesan, E., Chen, C. H., Snowdon, J. L., and Charnes, J. M., editors, *Winter Simulation Conference, BoD, San Diego*, volume 2, pages 1606–1610.

[Samsatli et al., 1998]Samsatli, N. J., Papageorgiou, L. G., and Shah, N. (1998). Robustness metrics for dynamic optimization models under parameter uncertainty. *AIChE Journal*, 44(9):1993–2006.

[Schmitt, 1885]Schmitt, R. (1885). Acetylsalicylic acid synthesis. *J. Prakt. Chem.*, 2 (31):397.

[Schnell_Publishing, 2004]Schnell_Publishing (10.5.2004). Chemical market reporter. *New York*.

[Shah and Pantelides, 1992]Shah, N. and Pantelides, C. C. (1992). Design of multipurpose batch plants with uncertain production requirements. *Industrial & Engineering Chemistry Research*, 31:1325–1337.

[Sparrow et al., 1975]Sparrow, R. E., Forder, G. J., and Rippin, D. W. T. (1975). The choice of equipment sizes for multiproduct batch plants. heuristics vs. branch and bound. *Industrial & Engineering Chemistry and Process Design Development*, 14:197–203.

[Stefanis et al., 1997]Stefanis, S. K., Livingston, A. G., and Pistikopoulos, E. N. (1997). Environmental impact considerations in the optimal design and scheduling of batch processes. *Computers & Chemical Engineering*, 21(10):1073–1094.

[Straub and Grossmann, 1993]Straub, D. A. and Grossmann, I. E. (1993). Design optimization of stochastic flexibility. *Computers & Chemical Engineering*, 17(4):339–354.

[Swaney and Grossmann, 1985]Swaney, R. E. and Grossmann, I. E. (1985). An index for operational flexibility in chemical process design. part 1: Formulation and theory. *AIChE Journal*, 31(4):621–630.

[Takamatsu et al., 1982]Takamatsu, T., Hashimoto, I., and Hasebe, S. (1982). Optimal design and operation of a batch process with intermediate storage tanks. *Industrial & Engineering Chemistry and Process Design Development*, 21:431–440.

[Terwiesch et al., 1994]Terwiesch, P., Agarwal, M., and Rippin, D. W. T. (1994). Batch unit optimization with imperfect modelling: A survey. *Journal of Process Control*, 4:238–246.

[van den Heever and Grossmann, 1999]van den Heever, S. A. and Grossmann, I. E. (1999). Disjunctive multiperiod optimization methods for design and planning of chemical process systems. *Computers & Chemical Engineering*, 23:1075–1095.

[Voudouris and Grossmann, 1996]Voudouris, V. T. and Grossmann, I. E. (1996). MILP model for scheduling and design of a special class of multipurpose batch plants. *Computers & Chemical Engineering*, 20(11):1335–1360.

[Wang et al., 1999]Wang, C., Quan, H., and Xu, X. (1999). Optimal design of multiproduct batch chemical processes using tabu search. *Computers & Chemical Engineering*, 23:427–437.

[Weisstein, 2006]Weisstein, E. W. (05. 2006). Nelder-mead method, mathworld - a wolfram web resource. *http://mathworld.wolfram.com/Nelder-Mead-Method.html.*

[Wellons and Reklaitis, 1989]Wellons, H. S. and Reklaitis, G. V. (1989). The design of multiproduct batch plants under uncertainty with staged expansion. *Computers & Chemical Engineering*, 13(1-2):115–126.

[Wellons and Reklaitis, 1991]Wellons, M. C. and Reklaitis, G. V. (1991). Scheduling of multipurpose batch chemical plants. 1. formulation of single-product campaigns. *Industrial & Engineering Chemistry Research*, 30:671–688.

[Wen and Chang, 1968]Wen, C. Y. and Chang, T. M. (1968). Optimal design of systems involving parameter uncertainty. *Industrial & Engineering Chemistry and Process Design Development*, 7(1):49–52.

[Yanofsky et al., 2006]Yanofsky, C., Taylor, D., and Thibault, J. (2006). Novel methodology for assessing a ranked pareto domain: Drift group analysis. *Chemical Engineering Science*, 61(4):1312–1320.

[Yeh and Reklaitis, 1987]Yeh, N. C. and Reklaitis, G. V. (1987). Synthesis and sizing of batch/semicontinuous processes: single product plants. *Computers & Chemical Engineering*, 11:639–654.

[Yi and Reklaitis, 2002]Yi, G. and Reklaitis, G. V. (2002). Optimal design of batch-storage network using periodic square wave model. *AIChE J.*, 48:1737–1753.

[Yoo et al., 1999]Yoo, D. J., Lee, H., Ryu, J., and Lee, I. (1999). Generalized retrofit design af multipurpose batch plants. *Computers & Chemical Engineering*, 23:683–695.

[Young and Reklaitis, 1989]Young, R. A. and Reklaitis, G. V. (1989). Capacity expansion study of a batch production line. *Ind. Eng. Chem. Res.*, 28:771–777.

[Zitzler et al., 2001]Zitzler, E., Deb, K., Thiele, L., and Coello, C. A. C. (1993-2001). *Evolutionary Multi-Criterion Optimization*. Lecture Notes in Computer Science. Springer, Zurich, Switzerland, First International Conference EMO 2001. 713677:1993.

Appendix A

A-1 Program heuristics

Table A-1: Equipment class matrix [EqClass] containing equipment classes used in the software.

Equipment class ID [EqClassID]	Equipment class [EqClass.Name]
1	Centrifuge
2	Column
3	Condenser
4	Filter - Cross Flow
5	Crystallizer
6	Extractor
9	Filter - Pot
10	Filter - Press
11	Filter - In-Line
14	Reactor
16	Dryer - Rotary
18	Filter - Sparkler
19	Tank
22	Dryer - Tray
24	Column - Continuous Packed
25	Column - Continuous Tray
28	Extractor - Continuous
29	Filter - Continuous
30	Heat Exchanger
32	Evaporator - Long Tube
34	Evaporator - Wiped Film
48	Centrifuge - Horizontal Basket
49	Centrifuge - Vertical Basket
50	Dryer
51	Filter - Dryer
52	Dryer - Agitated Pan
53	Hopper
54	Filter - Agitated Nutsche
55	Dryer - Conical
56	Dryer - Blender
57	Dryer - Horizontal Paddle
62	Filter - Tank Sheet
63	Reactor - Continuous
68	Centrifuge - Decanter
69	Filter - Depth
87	Filter - Bag
90	Dryer - Spray
91	Dryer - Fluid Bed - Continuous
92	Dryer - Freeze
135	Centrifuge - Filter
137	Heat Exchanger - Plate
138	Heat Exchanger - Shell and Tube
148	Dryer - Fluid Bed
175	Mixer - Static
178	Filter - Nutsche
179	Filter - Cartridge
256	Superclass

Table A-2: Operation classes matrix [OpClass] containing the operation class ID, name and scaling heuristics, which denotes the operation time scaling according to the operation volume: 0 - no scaling, 1 - linear scaling with volume, 2 - non-linear scaling with volume.

OpClassID	OpClass.Name	OpClass.Scaling
1	Age	0
2	Clean	0
3	QC-Test	0
4	Charge	0
7	Transfer	1
9	Transfer-Through-Heat-Exchanger	1
12	Elute-Column	0
14	Concentrate	1
15	Crystallize	0
16	Decant	0
17	Distill	1
18	Dry	1
19	Extract	0
20	Filter	0
22	Quench	0
24	Wash-Cake	0
25	Cool	1
26	Heat	1
27	Heat-To-Reflux-And-Age	0
28	Evacuate	0
29	Pressurize	0
30	Purge	0
31	Vent	0
32	React	0
35	pH-Adjust	0
38	Distill-Continuously	1
40	Extract-Continuously	1
47	Multi-Drop Centrifuge	2
48	Utilize	0
49	Yield React	0
51	Open-Close Vent	0
52	React-Distill	0
64	Mix	0
76	Pressure-Transfer	1

Table A-3: Class assignments matrix [A]. Operation classes are assigned to equipment classes. The operation classes reffer to Table A-2, the equipment classes to Table A-1. The operation class -1 assigned to superequipment class (ID 256) means that all operations are allowed in that equipment class. The marker -1 for equipment means all equipment is suitable for the given operation class.

A.OpClassID	A.EqClassID
1	14 19
2	-1
3	14 19
4	14 19
7	-1
9	30 137 138 14
12	2
14	14
15	5 14
16	6 14 19
17	2 24 25 3 14 34 32
18	16 22 50 52 55 56 57 90 92 148
19	6 14
20	4 9 10 11 18 29 51 54 62 69 84 87 89
22	14
24	1 4 9 10 11 18 29 48 49 51 54 62 68 69 71 76 81 84 87 89 132
25	14 19
26	14 19
27	3
28	14 19
29	14 19
30	14 19
31	14 19
32	14
35	14 19
38	24 25
40	28
47	1 48 49 68 71 76 81 135
48	-1
49	14
51	14 19
52	3
64	14 175 19
76	14 19
83	148
-1	256

A-2 XML data for the cost function

Product Cost Data Overview ©2005 A.M.

Project description: Vitamin C

Author of xml: Andrej Mosat

Date of cost data: 5.03.2005

Campaign size projected: 1000000 kg

Projected selling price of product: 10 CHF/kg

1 Costs independent from design/production line

Material	L-Sorbose 2.34 CHF/kg amount fixed: 1295 t per: campaign	Sulfuric Acid 125 CHF/t amount fixed: 3071 t per: campaign	NaOH 0.156 CHF/kg amount fixed: 1036 t per: campaign
	Acetone 2.08 CHF/kg amount fixed: 1271.2 t per: campaign	Toluene 1.356 CHF/kg amount fixed: 276 t per: campaign	Water 0.005 CHF/kg amount fixed:9620 t per: campaign
	Sodium-Hypochlo-rite 0.61 CHF/kg amount fixed: 388.5 t per: campaign	Ethanol 0.932 CHF/kg amount fixed: 74 t per: campaign	
Waste management	Waste and other materials 0.0093 CHF/kg amount fixed: 14800 t per: campaign		

2 Costs dependent from design/production line

2.1 Energy and utilities

plantC4	plantC4ce	plantC4r	plantC4r-ce	SuperPlant
Steam	*Steam*	*Steam*	*Steam*	*Steam*
45.34 CHF/ MWh	45.34 CHF/ MWh	45.34 CHF/ MWh	45.34 CHF/ MWh	45.34 CHF/MWh
amount fixed: 1.5 MWh amount variable: 0.3 MWh per: nrreact/ct	amount fixed: 1.5 MWh amount variable: 0.3 MWh per: nrreact/ct	amount fixed: 1.5 MWh amount variable: 0.3 MWh per: nrre-act/ct	amount fixed: 1.5 MWh amount variable: 0.3 MWh per: nrreact/ct	amount fixed: 1.5 MWh amount variable: 0.3 MWh per: nrreact/ct
Electricity	*Electricity*	*Electricity*	*Electricity*	*Electricity*
77.59 CHF/MWh	77.59 CHF/MWh	77.59 CHF/MWh	77.59 CHF/MWh	77.59 CHF/MWh
amount fixed: 1.5 MWh	amount fixed: 1.5 MWh	amount fixed: 1.5 MWh	amount fixed: 1.5 MWh	amount fixed: 1.5 MWh
Nitrogen	*Nitrogen*	*Nitrogen*	*Nitrogen*	*Nitrogen*
1.91 CHF/m^3	1.91 CHF/m^3	1.91 CHF/m^3	1.91 CHF/m^3	1.91 CHF/m^3
amount fixed: 0.5 m^3	amount fixed: 0.5 m^3	amount fixed: 0.5 m^3	amount fixed: 0.5 m^3	amount fixed: 0.5 m^3

2.2 Labour cost

plantC4	plantC4ce	plantC4r	plantC4r-ce	SuperPlant
40 CHF/hr amount fixed: 1 hr per: 1/ct amount variable: 0.5 hr per: nreq	40 CHF/hr amount fixed: 1 hr per: 1/ct amount variable: 0.5 hr per: nreq	40 CHF/hr amount fixed: 1 hr per: 1/ct amount variable: 0.5 hr per: nreq	40 CHF/hr amount fixed: 1 hr per: 1/ct amount variable: 0.5 hr per: nreq	40 CHF/hr amount fixed: 1 hr per: 1/ct amount variable: 0.5 hr per: nreq

2.3 Plant rent

plantC4	plantC4ce	plantC4r	plantC4r-ce	SuperPlant
10 CHF/hr amount fixed: 1 hr per: 1/ct amount variable: 0.5 hr per: nreq	10.5 CHF/hr amount fixed: 1 hr per: 1/ct amount variable: 0.5 hr per: nreq	10.7 CHF/hr amount fixed: 1 hr per: 1/ct amount variable: 0.5 hr per: nreq	11.2 CHF/hr amount fixed: 1 hr per: 1/ct amount variable: 0.5 hr per: nreq	9 CHF/hr amount fixed: 1 hr per: 1/ct

2.4 Changeover

plantC4	plantC4ce	plantC4r	plantC4r-ce	SuperPlant
102 CHF/hr amount fixed: 0.5 hr	102 CHF/hr amount fixed: 0.5 hr	102 CHF/hr amount fixed: 0.5 hr	102 CHF/hr amount fixed: 0.5 hr	102 CHF/hr amount fixed: 0.5 hr

2.5 Other

plantC4	plantC4ce	plantC4r	plantC4r-ce	SuperPlant
10 CHF amount fixed: 1 const	10 CHF amount fixed: 1 const	10 CHF amount fixed: 1 const	10 CHF amount fixed: 1 const	10 CHF amount fixed: 1 const

2.6 Investment

plantC4	plantC4ce	plantC4r	plantC4r-ce	SuperPlant
0 CHF	720000 CHF	510000 CHF	1300000 CHF	500000 CHF

Legend: each parameter has a unit and value field, for example: unit: CHF, value: 10. In general, the field consists of 'fixed amount' and 'variable amount'. The basis of computation is indicated by the tag: 'per'. For example: waste material is displayed as: amount fixed: 14800 t, per: campaign, which means that this amount is counted for the whole campagin. Note: if the field name "per:" is misss-ing in the description of "fixed" or "variable" amount, it means: "per: batch" or accordingly "per:campaign" depending on context, this sheet is an overview, not the equation or formula itself. If the "per: " field is stated as: "per: XY" it means: "per: XY and batch" The abbreviation const means a constant amount.

For additional information about the field names, see Figure 2-9.

A-3 Vitamin C Reichstein synthesis in Batch Plus™ software

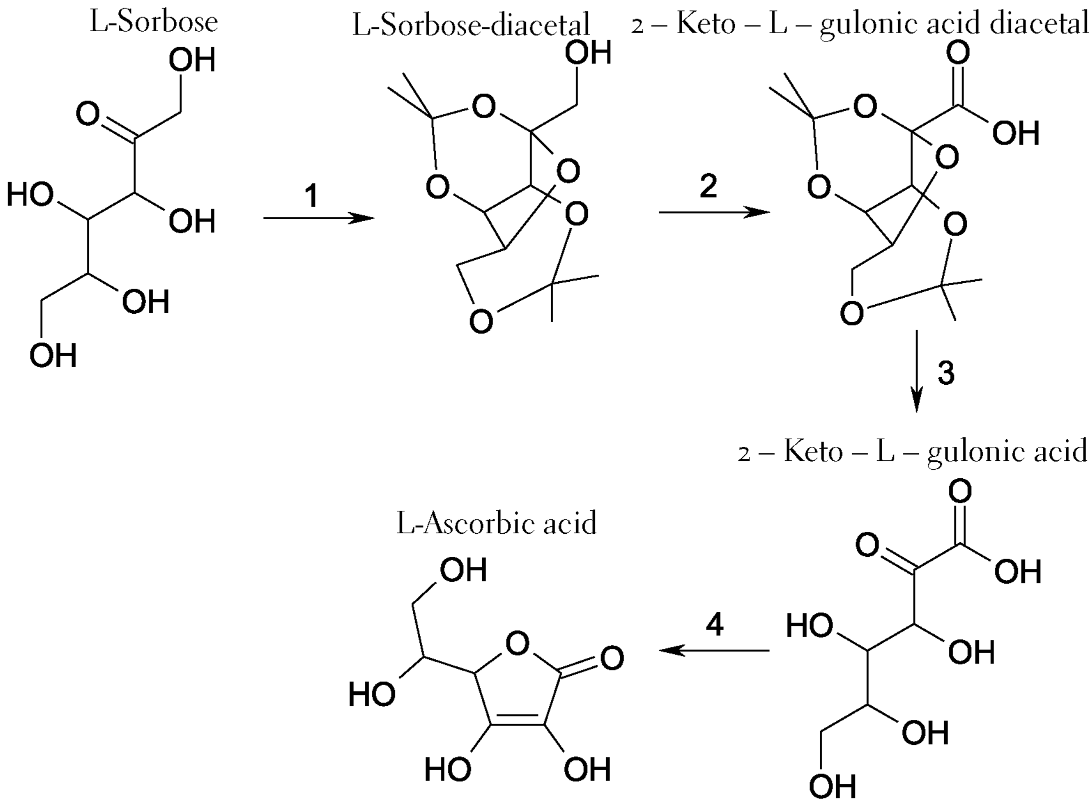


Figure A-1: Vitamin C reaction scheme.

Step: Vitamin C Synthesis

----

1. **L-Sorbose -> L-Sorbose Diacetal**
   Comments: Diacetal-Protection group for later oxidation to Carbonic acid.

1.1. Charge R16b3 with 426 kg of L-Sorbose. The charge time is 30 min. Charge R16b3 with 8540 liter of ACETONE. The feed rate is 10 Cubic m/h. Charge R16b3 with 340 liter ofSULFURIC-ACID. The feed rate is 10 Cubic m/h. Dissolve 100% of all solids.

1.2. Cool unit R16b3 to 4 C. The cooling time is 1 h.

1.3. React in unit R16b3 via reaction1. Reaction occurs over 5 h.

1.4. Charge R16b3 with 340 kg of SODIUM-HYDROXIDE. The charge time is 30 min. Charge R16b3 with 855 liter of WATER. The feed rate is 15 Cubic m/h. Dissolve 100% of all solids.

1.5. Distil the batch in unit R16b3. The operation time is 2 h. The overhead, named Aceton-Recycle, is sent to T10a2. The process condenser is Co10b3. Separation is: 95% of ACETONE goes to Overhead and 5% of WATER goes to Overhead. Unspecified materials go to Bottoms.

1.6. Charge R16b3 with 1280 liter of TOLUENE. The feed rate is 15 Cubic m/h.

1.7. Extract in unit R16b3 over 90 min. Partition coefficient(mass basi top/bottom) of L-Sorbose is 3, of Diacetone-L-Sorbose is 3, of TOLUENE is 100, of WATER is 0.01 and of ACETONE is 1. Unspecified materials go to Bottom. The bottom layer is sent to T16a3.
The transfer rate is 15 Cubic m/h.

----

2. L-Sorbose Diac -> 2-Keto-L-Gulonic acid Diacetal

Comments: oxidation to gulonic acid under the protection of acetal-protection groups

2.1. Transfer contents of unit R16b3 to R10c1. The flow rate is 15 Cubic m/h.
2.2. Charge R10c1 with 128 kg of SODIUM-HYPOCHLORITE. The charge time is 30 min. Charge R10c1with 4.3 kg of NICKEL-SULFATE. The charge time is 1 min. Dissolve 100% ofall solids.
2.3. Heat unit R10c1 to 60 °C. The heating time is 1 h.
2.4. React in unit R10c1 via reaction2. Reaction occurs over 3 h.
2.5. Charge R10c1 with 213 liter of SULFURIC-ACID. The feedrate is 15 Cubic m/h. Charge R10c1 with 850 liter of ETHANOL. The feed rate is 15 Cubic m/h. Maintain the temperature at 60 C.
2.6. Transfer contents of unit R10c1 to Cr16a2. The flow rate is 15 Cubic m/h.
2.7. Crystallize the batch in unit Cr16a2. The followingcomponents are separated in the crystal phase: 97% of 2-keto-gluconic acid diacetal, 10% of Diacetone-L-Sorbose,10% of L-Sorbose and 0.1% of ETHANOL. The crystallizationtime is 2.5 h.
2.8. Transfer contents of unit Cr16a2 to R16d4. The flow rate is 15 Cubic m/h.

----

3. 2-Keto-L-Gulonic acid diac -> 2-Keto-L-Gulonic acid

Comments: Removal of acetal-protection group by hydratation

3.1. Charge R16d4 with 430 liter of WATER. The feed rate is 15 Cubic m/h. Maintain the temperature at 90 C. Dissolve 100% of all solids.
3.2. Heat unit R16d4 to 100 °C. The heating time is 1 h.
3.3. React in unit R16d4 via reaction3. Reaction occurs over 1 h.
3.4. Transfer contents of unit R16d4 to Cr10a2. The flow rate is15 Cubic m/h.
3.5. Crystallize the batch in unit Cr10a2. The followingcomponents are separated in the crystal phase: 97% of 2-Keto-L-glutonic acid, 30% of 2-keto-gluconic aciddiacetal, 10% of Diacetone-L-Sorbose and 10% of L-Sorbose.The crystallization time is 1 h.
3.6. Purge unit R10e1. Purge 1 times with NITROGEN for 30 min each.
3.7. Transfer contents of unit Cr10a2 to R10e1. The flow rate is15 Cubic m/h.

----

4. 2-Keto-L-Gulonic acid -> L-Ascorbic acid

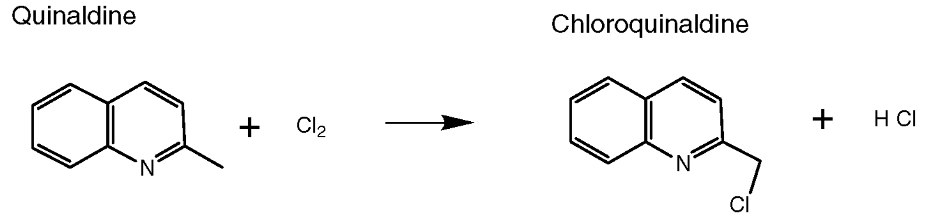
Comments: cyclisation, acidic catalyse with HCl

4.1. Charge R10e1 with 6403 liter of TOLUENE. The feed rate is15 Cubic m/h. Charge R10e1 with 430 liter of HYDROGEN-CHLORIDE. The feed rate is 15 Cubic m/h. Dissolve100% of all solids.
4.2. React in unit R10e1 via reaction4. Reaction occurs over 1 h.
4.3. Centrifuge the batch from unit R10e1 in centrifuge Ce1.6a2.The number of drops is 8. The cake is s ent to Cr10b4. Thecake discharge flow rate is 5 kg/s.
4.4. Charge Cr10b4 with 2130 liter of WATER. The feed rate is15 Cubic m/h. Dissolve 100% of all solids.
4.5. Crystallize the batch in unit Cr10b4. The followingcomponents are separated in the crystal phase: 98% of ASCORBIC-ACID, 5% of Diacetone-L-Sorbose, 5% of L-Sorbose,5% of 2-Keto-L-glutonic acid and 5% of 2-keto-gluconic aciddiacetal. The crystallization time is 90 min.
4.6. Transfer contents of unit Cr10b4 to T16b3. The flow rate is15 Cubic m/h.

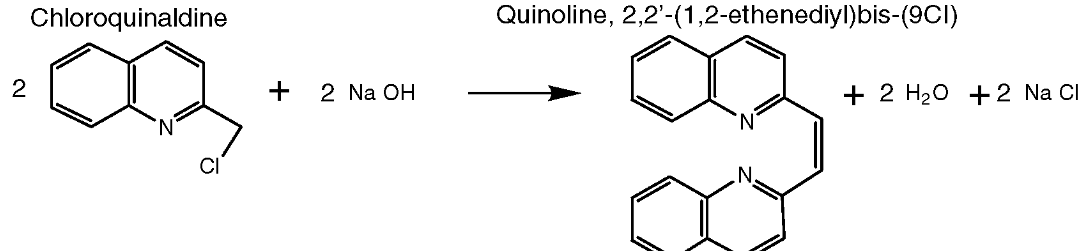
A-4 Quinaldine derivate synthesis - Product H

A reaction scheme

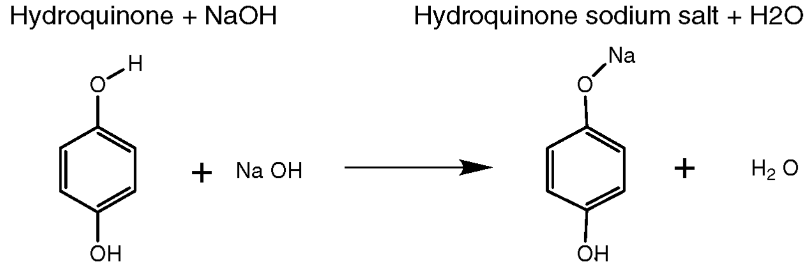
1) Chlorination



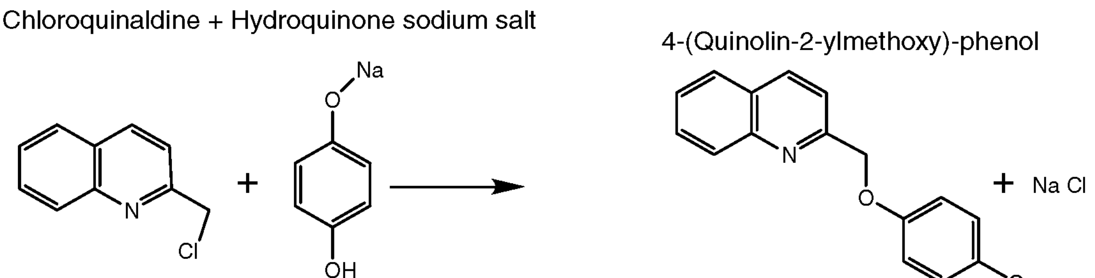
3b) Condensation reactions (Byproduct/Impurity formation)



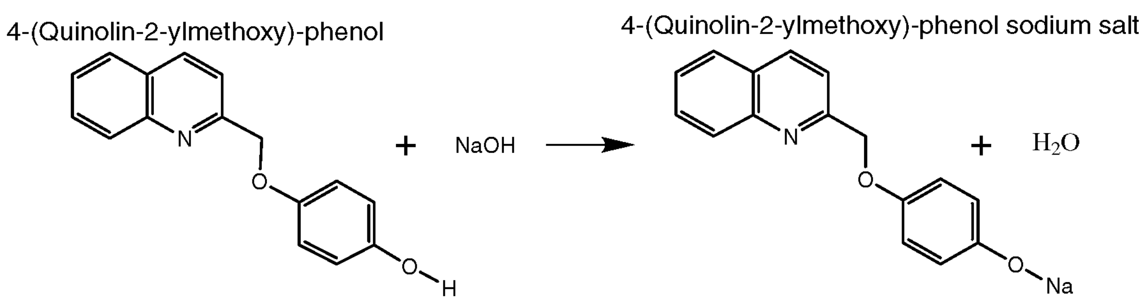
3c) Condensation reactions



3d) Condensation reactions



4) Solubilization



1) Chlorination Reaction and Salt Formation

The first reaction step involves the chlorination of quinaldine. Quinaldine is dissolved in carbon tetrachloride (CCl_4) and reacts with gaseous Cl_2. The yield of the reaction is around 98%. The generated HCl is neutralized using Na_2CO_3. The stoichiometry and yield of the three reactions follows:

Quinaldine + Cl_2 ===> Chloroquinaldine + HCl Yield = 98 %

Na_2CO_3 + HCl ===> $NaHCO_3$ + NaCl Yield = 100 %

$NaHCO_3$ + HCl ===> NaCl + H_2O + CO_2 Yield = 100 %

2) The second reaction

(reaction not displayed in the reaction scheme) involves the formation of Chloroquinaldine.HCl. The added HCl first neutralizes the remaining $NaHCO_3$ and then reacts with chloroquinaldine to form its salt. The stoichiometry and yield of the two reactions follows:

$NaHCO_3$ + HCl ===> NaCl + H_2O + CO_2 Yield = 100 %

Chloroquinaldine + HCl ===> Chloroquinaldine.HCl Yield = 100 %

Small amounts of generated CO_2 and volatilized CCl_4 are vented. The presence of water (added with HCl as hydrochloric acid solution) and CCl_4 leads to the formation of two liquid phases. The small amounts of unreacted quinaldine and chloroquinaldine remain in the organic phase while the salts Chloroquinaldine.HCl and NaCl move to the aqueous phase.

3) Condensation Reactions

The third reaction step involves the condensation of chloroquinaldine and hydroquinone. First, the salt chloroquinaldine.HCl is converted back to chloroquinaldine using NaOH. Then, hydroquinone reacts with NaOH and yields hydroquinone.Na. Finally, chloroquinaldine and hydroquinone.Na react and yield the desired intermediate product. Along with product formation, a small amount of chloroquinaldine dimerizes and forms an undesirable by-product (Impurity) that needs to be removed from the product. The stoichiometry and yield of the four reactions follows:

Chloroquinaldine.HCl + NaOH ===> Chloroquinaldine + NaCl + H_2O Yield = 100 %

2 Chloroquinaldine + 2 NaOH ===> 2 H_2O + 2 NaCl + Byproduct/Impurity Yield = 2 %

Hydroquinone + NaOH ===> Hydroquinone .Na + H_2O Yield = 100 %

Chloroquinaldine + Hydroquinone.Na ===> Product + NaCl Yield = 100 %

4) Solubilization Reaction

The Product/Impurity cake recovered by filtration is added into a NaOH solution. The Product molecules react with NaOH forming Product.Na which is soluble in water. The Impurity molecules remain in solid phase. The stoichiometry and yield of the solubilization reaction follows:

Product + NaOH ===> Product.Na + H_2O Yield = 100 %

5) Precipitation Reaction Step

The excess NaOH is neutralized using HCl and then Product.Na is converted back to Product. The stoichiometry and yield of the two reactions follows:

HCl + NaOH ===> H_2O + NaCl Yield = 100 %

HCl + Product .Na ===> Product + NaCl Yield = 100 %

Batch Plus recipe

1. Chlorination

1.1. Charge r6-1a with 1300 kg of CARBON-TETRACHLORIDE. The feed rate is 9 Cubic m/h. Use 10 °C Cooling Water on the Entire Jacket at a rate of 1 Cubic m/h.
1.2. Charge r6-1a with 390 kg of QUINALDINE. The charge time is 30 min.
1.3. Charge r6-1a with 275 kg of SODIUM-BICARBONATE. The charge time is 30 min.
1.4. Close the vent in unit r6-1a.
1.5. React in unit r6-1a via 1-chlorination. Reaction occurs over 8.7 h. The final temperature of the batch is 50 °C. Continuously add 236 kg of CHLORINE. The feed time is 6 h. The feed begins 10 min after the operation start time. Extra time for this Operation is 10 min.
1.6. QC-Test the material in unit r6-1a. Continue operation while waiting for the test. The expected test time is 60 min.
1.7. Cool unit r6-1a to 25 °C. The cooling time is 65 min. Use 10 °C Cooling Water at a rate of 3.25 Cubic m/h.
1.8. Close the vent in unit r6-1a.
1.9. Evacuate r6-1a to 0.1 bar. Evacuation time is 20 min.
1.10. Pressurize unit r6-1a to pressure 1 atm. . Pressurization time is 10 min.

2. Salt formation

2.1. Charge r6-1a with 180 kg of CARBON-TETRACHLORIDE. The feed rate is 9 Cubic m/h. Maintain the temperature at 25 °C.
2.2. Charge r6-1a with 2710 kg of WATER. The feed rate is 9 Cubic m/h.
2.3. Charge r6-1a with 120 kg of HCl-100%-l. The feed rate is 2 Cubic m/h.
2.4. React in unit r6-1a via 2-salt-formation. Reaction occurs over 70 min. The final temperature of the batch is 50 °C. There is no emission control for this operation. Extra time for this Operation is 10 min.
2.5. Close the vent in unit r6-1a.
2.6. Evacuate r6-1a to 0.1 atm. Evacuation time is 20 min.
2.7. Pressurize unit r6-1a to pressure 1 atm. . Pressurization time is 20 min.
2.8. Open the vent in unit r6-1a.
2.9. Extract in unit r6-1a over 90 min. Separation is: 100% of WATER goes to Bottom, 100% of QUINALDINE goes to Top, 100% of Chloroquinaldine-HCl goes to Bottom and 100% of SODIUM-CHLORIDE goes to Bottom. Unspecified materials go to Top. The bottom layer, named main, is sent to r6-2a. The transfer time is 15 min. There is no emission control for this operation.
2.10. Open the vent in unit r6-2a.

3. Condensation

3.1. Charge r6-2a with 280 kg of WATER. The charge time is 15 min.
3.2. Charge r6-2a with 449 kg of Hydroquinone. The charge time is 30 min.
3.3. Charge r6-2a with 1450 kg of METHANOL. The feed rate is 8 Cubic m/h.
3.4. Charge r6-2a with 280 kg of SODIUM-HYDROXIDE. The charge time is 25 min. Dissolve 100% of all solids.
3.5. React in unit r6-2a via 3-Condensation. Reaction occurs over 15 h. The final temperature of the batch is 50 °C. Extra time for this Operation is 10 min.

3.6. QC-Test the material in unit r6-2a. Continue operation while waiting for the test. The expected test time is 60 min.

4. Solubilization

4.1. Filter the batch from unit r6-2a in filter filter-nutsche-4-a. The slurry flowrate is 1.1 Cubic m/h. The cake contains 100% of 4quinolin-methoxyphenol in the Solid phase and 100% of Chloroquinaldine-impurity in the Solid phase. The cake contains 20% of WATER in the Liquid phase. Extra time for this Operation is 20 min.
4.2. Wash the cake in unit filter-nutsche-4-a. For each wash, use 2450 kg of WATER. The feed rate is 2 Cubic m/h. The moisture content in the final cake is 20%.
4.4. Transfer contents of unit filter-nutsche-4-a to r6-3a. Extra time for this Operation is 5 min.
4.5. Charge r6-3a with 4100 kg of WATER. The charge time is 30 min.
4.6. Charge r6-3a with 195 kg of NaOH (s). The charge time is 20 min.
4.7. React in unit r6-3a via 4-Solubilization. The reaction is Adiabatic. Reaction occurs over 90 min.
4.9. Filter the batch from unit r6-3a in filter filter-nutsche-4b. The slurry flowrate is 1.1 Cubic m/h. The mother liquor is sent to r6-4a. The filter separates 100% of all solids. The moisture content in the final cake is 5%.
4.10. Charge r6-4a with 543 kg of HCl (36%). The feed rate is 8 Cubic m/h.
4.11. React in unit r6-4a via 5-Precipitation-final. The reaction is Adiabatic. Reaction occurs over 30 min.
4.12. Filter the batch from unit r6-4a in filter filter-nutsche-4c. The slurry flowrate is 1.1 Cubic m/h. The cake contains 100% of 4quinolin-methoxyphenol in the Solid phase. The cake contains 20% of WATER in the Liquid phase. Extra time for this Operation is 20 min.
4.13. Wash the cake in unit filter-nutsche-4c. For each wash, use 2446 kg of WATER. Wash the cake 1 times.
4.14. Transfer contents of unit filter-nutsche-4c to r10-1a. Extra time for this Operation is 5 min.
4.16. Charge r10-1a with 4320 kg of ISOPROPYL-ALCOHOL. The feed rate is 9 Cubic m/h. Dissolve 100% of all solids. Extra time for this Operation is 30 min.

5 Product recovery

5.1. Age the contents of unit r10-1a for 3.5 h. Comments: This is the filtration through GAF Filter
5.2. Transfer contents of unit r10-1a to cryst10-a1.
5.3. Crystallize the batch in unit cryst10-a1. The following components are separated in the crystal phase: 97% of 4quinolin-methoxyphenol. The crystallization time is 4 h.
5.4. Centrifuge the batch from unit cryst10-a1 in centrifuge centrifuge-800b. The mass of wet cake per drop is 120 kg. The slurry flow rate is 1 Cubic m/h. The speed of rotation of the centrifuge for slurry charge is 10000 Rev/min. The cake moisture represents 30% of the total cake. The cake named product-crystallized is sent to dry5.3-a2. The cake discharge time is 12 min.
5.5. Dry the batch in unit dry5.3-a2. The drying time is 12 h. The drying temperature is 90 °C. The drying pressure is 0.8 atm. The moisture content in the final cake is 3%. The product recovery is 99.5%.

A-5 Acetylsalicylic acid reaction scheme, Kolbe-Schmitt synthesis

This reaction scheme is used in the case study 3. 2.3 *New plant design for selected recipe - Acetylsalicylic Acid case study* and uses a Kolbe-Schmitt process (Schmitt, 1885, Lindsey and Jeskey, 1957, Kolbe, 1860). Acetylsalicylic acid (Aspirin, molar weight 180.16 g/mol) is commercially synthesized using a two-step process. First, phenol is treated with a sodium base generating sodium phenoxide, which is then reacted with carbon dioxide under high temperature and pressure to yield salicylate (Reaction 1, yield 80 %). Salicylate which is acidifed yields salicylic acid (Reaction 2, yield 100 %). Salicylic acid is then acetylated using acetic anhydride, yielding Aspirin and acetic acid as a byproduct (Reaction 3, yield 95 %). As a final step, conversion to sodium salt is performed (Reaction 4, yield 100 %).

Formulations containing high concentrations of acetylsalicylic acid can undergo autocatalytic degradation to salicylic acid in moist conditions, yielding salicylic acid and acetic acid.

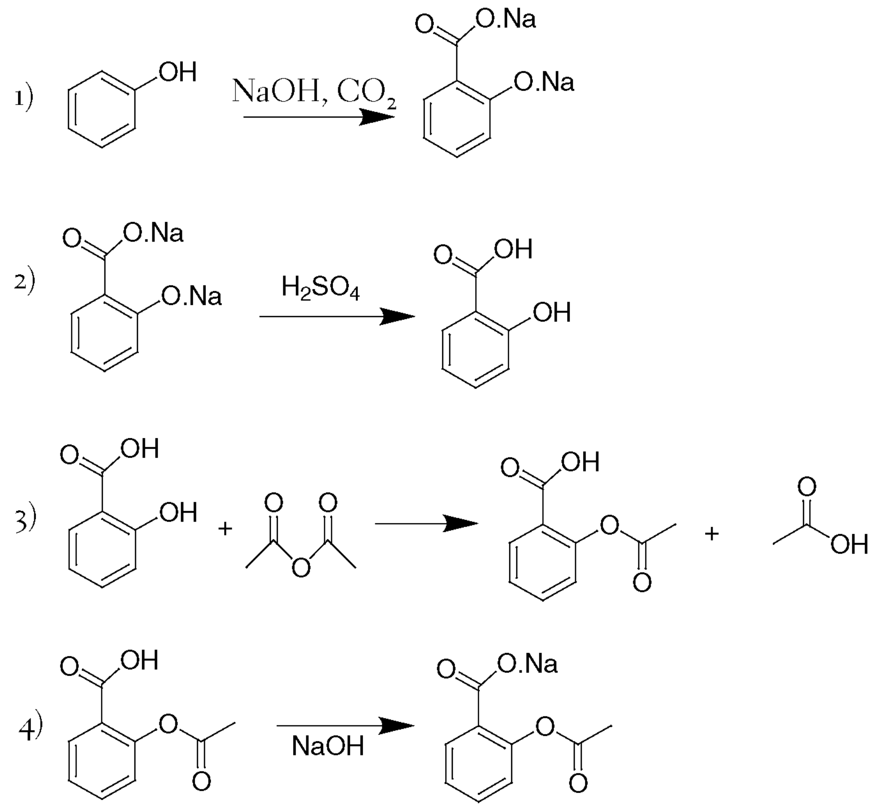


Figure A-2: Kolbe-Schmitt Synthesis of Aspirin.

A-6 Acetylsalicylic acid base recipe, one-step reaction

Reaction scheme

The reaction scheme presented by (Dimmer, 1999, BatchPlus(TM), 2002) consists of one reaction step (see Figure 4-6) with 93% yield at a temperature 90° C.

	Salicylic acid	Acetic anhydride	Acetylsalicylic acid	Acetic acid
Molecular weight [g/mol]:	138	102	180	60

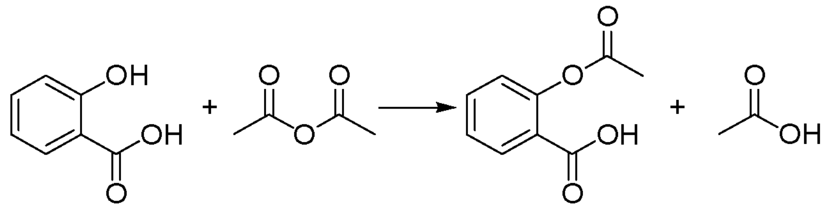


Figure A-3: Aspirin (Acetylsalicylic acid) synthesis.

The raw materials are Salicylic acid and Acetic anhydride, while Acetic acid is used as solvent, and is also a by-product.

Batch Plus recipe

1. Aspirin synthesis

1.1. Charge R35 with 4548 kg of SALICYLIC-ACID. The charge time is 15 min.
1.2. Charge R35 with 4708 kg of ACETIC-ANHYDRIDE. The charge time is 15 min.
1.3. Charge R35 with 1976 kg of ACETIC-ACID. The charge time is 15 min. Dissolve 100% of all solids.
1.4. React in unit R35 via Aspirin Sythesis. Reaction occurs over 150 min. The final temperature of the batch is 75 °C.
1.5. Transfer contents of unit R35 to R33. The transfer time is 15 min.
1.6. Continue Reaction Utilize unit R33 for 150 min.
1.7. Transfer contents of unit R33 to R32. The transfer time is 15 min.
1.8. Continue Reaction Utilize unit R32 for 150 min.

2. Crystallization of ACETYLSALICYLIC-ACID

2.1. Transfer contents of unit R32 to R23. The transfer time is 15 min.
2.2. Crystallize the batch in unit R23. The following components are separated in the crystal phase: 100% of ACETYLSALICYLIC-ACID. The crystallization time is 210 min.
2.3. Transfer contents of unit R23 to R80. The transfer time is 15 min.

3. Centrifugation

3.1. Centrifuge the batch from unit R80 in centrifuge C5 & C3. Transfer 100% of the batch to the centrifuge. The slurry feed time is 15 min. The deliquoring time is 40 min. The centrifuge retains 100% of the solid. The cake moisture represents 5% of the total cake. The mother liquor is sent to T8. The cake is sent to R8. The cake discharge time is 15 min.

4. Drying

4.1. Transfer contents of unit R8 to D60. The transfer time is 15 min.
4.2. Dry the batch in unit D60. The drying time is 120 min. The drying temperature is 100 °C.
4.3. Transfer contents of unit D60 to Big-bag. This is the key step output. The transfer time is 15 min.

5. Distillation

5.1. Transfer contents of unit T8 to R22. The transfer time is 15 min.
5.2. Distill the batch in unit R22. The operation time is 180 min. Separation is: 90% of ACETIC-ACID goes to Overhead and 90% of ACETIC-ANHYDRIDE goes to Overhead. Unspecified materials go to Bottoms.

Curriculum Vitae

Andrej Mošať

born on December 14th, 1978 in Skalica
Citizen of Slovak Republic

Education

2003–2006	Doctor of sciences (Ph.D.) in computer aided process design ETH Zurich, Institute for Chemical and Bioengineering, Safety and Environmental Technology Group, Prof. K. Hungerbühler, *Deterministic and stochastic batch design optimization techniques*
2002	Master of Science in Technical Chemistry (Ing.), Chemical Process Engineering at Slovak Technical University Bratislava, Faculty of Chemical and Food Technology
1997–2002	studies at Slovak Technical University Bratislava, Faculty of Chemical and Food Technology
2000	Bachelor's Degree in Technical Chemistry (Bc.), STU Bratislava
1993–1997	studies at Gymnasium Senica, passed with grade "very good"

Languages

Slovak and Czech language, English - fluently, German - fluently, French - basic

Computer Sciences

Linux and Windows Administration. Programming: Matlab, Python, Java, C, PHP4, Bash, VBA, Basic, SQL, Pascal, XML/XPATH/XSLT. Software skills: CFD Fluent, Femlab, Aspen Engineering Suite, Batch Plus, Batch Pro, Maple, Origin, Office, all major DTP and graphic packages, Apache, MySQL, DB2, Oracle, TopicMaps, technical XML processing, ...

Professional experience

2003–2006	ETH Zurich, teaching assistence in subjects *Chemical engineering, Chemistry*
2003	Biotika a.s. *Slovenská Ľupča*, Energy savings audit and consulting project, cooperation with STU Bratislava
2002	Socrates/Erasmus student at Technical University Vienna, Computational fluid dynamic simulation of artificial kidney
2001–2002	Energy savings audit and consulting in factories across Slovakia in cooperation with Doc. Ing. Otto Mierka, CSc.
1998–1999	Computer administrator and programmer at Elektroakustika, š.p. v konkurze
1997–ongoing	Various IT consulting and problem solving contracts

Side activities

2001–2005	Business activities; awarded "The most original business plan" in SAEF Venture competition Bratislava 2001, participant of the Venture 2005 Zurich
2001	BEST Summer Course Copenhagen - Creating Sustainable Products through Radical Innovation, Danish Technical University Copenhagen www.dtu.dk, Prof. Tim McAloone
2000	BEST Summer Course Bordeaux - Composite Materials, ENSAM Bordeaux , France
2000	Marketing & Kommunikation course certificate by Prof. Stephan Wolf, Hamburg

Interests

Photography, graphic design, computer, travelling, trekking, outdoors, summer camps for children, constructing and flying aeroplane models, ...