Strategic Biopharmaceutical Production Planning for Batch and Perfusion Processes

A thesis submitted to UCL

for the degree of

Engineering Doctorate

by

Cyrus Carlos Siganporia

The Advanced Centre for Biochemical Engineering
Department of Biochemical Engineering
UCL
Torrington Place
London WC1E 7JE

July, 2016

Declaration

I, Cyrus Siganporia, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the thesis.

Abstract

Capacity planning for multiple biopharmaceutical therapeutics across a large network of manufacturing facilities, including contract manufacturers, is a complex task. Production planning is further complicated by portfolios of products requiring different modes of manufacture: batch and continuous. Capacity planning decisions each have their own costs and risks which must be carefully considered when determining manufacturing schedules. Hence, this work describes a framework which can assimilate various input data and provide intelligent capacity planning solutions.

First of all, a mathematical model was created with the objective of minimising total cost. Various challenges surrounding the biomanufacturing of both perfusion and fed-batch products were solved. Sequence-dependent changeover times and full decoupling between upstream and downstream production suites were incorporated into the mixed integer linear program, which was used on an industrial case study to determine optimal manufacturing schedules and capital expenditure requirements. The effect of varying demands and fermentation titres was investigated via scenario analysis. To improve computational performance of the model, a rolling time horizon was introduced, and was shown to not only improve performance but also solution quality.

The performance of the model was then improved via appropriate reformulations which consider the state task network (STN) topology of the problem domain. Two industrial case studies were used to demonstrate the merits of using the new formulation, and results showed that the STN improved performance in all test cases, and even performed better than the rolling time horizon

approach from the previous model in one test case. Various strategic options regarding capacity expansion were analysed, in addition to an illustration of how the framework could be used to de-bottleneck existing capacity issues.

Finally, a multi-objective component is added to the model, enabling the consideration of strategic multi-criteria decision making. The ϵ-constraint method was shown to be the superior multi-objective technique, and was used to demonstrate how uncertain input parameters could affect the different objectives and capacity plans in question.

Acknowledgements

I would like to thank my academic supervisor, Prof. Suzanne Farid, for her continual support, trust and insightful guidance throughout the course of my project. I would also like to show my appreciation to Prof. Lazaros Papageorgiou, who has provided fruitful discussions and directions surrounding my thesis. I wish to extend my gratitude to Bayer Technology Services, in particular Soumitra Ghosh, Christian Rämsch, Andreas Schluck, Brijesh Tanjore Vasudeva Rao, Todd Roman, Scott Probst, and Thomas Daszkowski. Their time, supervision and industrial knowledge was immensely useful in developing this work.

I wish to thank the UK Engineering and Physical Sciences Research Council (EPSRC) Centre for Innovative Manufacturing in Emergent Macromolecular Therapies, hosted by UCL with Imperial College London and a consortium of industrial and government users. Financial support from the EPSRC and Bayer Technology Services is gratefully acknowledged.

I wish to extend my thanks to friends who have helped along the way, including Songsong Liu and Richard Allmendinger. Finally, I would like to thank my family for their patience and kindness throughout my doctorate degree.

Contents

Chapter 1

Literature Review

The biopharmaceutical industry has grown enormously since the first drug was released to the market in 1982. In the year 2000, there were 84 biopharmaceuticals approved globally, and by 2014 that number had grown to almost 250 (Walsh, 2014). However, this number may be closer to 170, since some of the therapeutics are very similar to each other biologically. This rapid growth is largely down to advances in molecular biology technology, providing improved platforms for the discovery and manufacture of monoclonal antibodies, protein hormones and genetically engineered vaccines (three major biopharmaceuticals). The success of these drugs can be measured by the profitability and growth of the companies manufacturing them. In 2014 alone, revenue for biopharmaceutical companies within the US, Europe, Canada and Australia increased by 24% (Ernst & Young, 2015). However, these biopharmaceutical drugs take approximately 8 years to go from initial development to reaching the market, placing huge pressures on the companies to reduce development and manufacturing costs (Foo et al., 2001). This, along with the inherent risks associated with biopharmaceutical sector, provides the reasoning behind the development of a decision support tool to help the industry perform more efficiently under uncertain conditions.

This chapter will discuss the development process of new drugs, and the pressures facing the biopharmaceutical industry. It will also describe some work that has already been carried out on capacity planning, and explain some of the techniques used in optimisation.

1.1 Biopharmaceutical Drug Development and Manufacturing

In order to get a drug to the market, it must first undergo preclinical and clinical trials, and then if successful, a New Drug Application (NDA) can be applied for and the drug then sold to the market. However, many drug candidates will be unsuccessful, and thus biopharmaceutical companies must develop many drug candidates simultaneously so that hopefully at least one will succeed. In general, only 1 in every 5,000 to 10,000 molecules that enter the drug discovery stage will successfully reach the market (Lipsky and Sharp, 2001), and on average it takes 8-12 years and has been estimated to cost between $1 - 1.8 billion (Adams and Brantner, 2010; Paul et al., 2010). The drug discovery stage involves computational chemistry, which is followed by 2-4 years of preclinical studies on animals. If successful, an investigational new drug (IND) application can be opened with the Food and Drug Administration (FDA), and then clinical trials on humans can begin. Phase I, II, and III take approximately one, two, and three years respectively to complete, and finally the manufacturer files for an NDA with the FDA for approval. Sometimes the FDA requires further studies to be undertaken before approval can be granted. Even after granting approval they can ask the manufacturer to continue post-marketing studies, especially for drugs which are administered over long periods. Figure 1.1 and Table 1.1 highlight some of the costs and risks involved in biopharmaceutical drug development, and the duration for each stage. The data was collated by Nie from work published by Paul et al. (2010) and DiMasi and Grabowski (2007).

During the preclinical and clinical trials, material must obviously be manufactured, but at a smaller scale than commercial production. There are two main upstream processes in use today for mammalian cell culture - fed-batch and perfusion. In fed-batch mode, media and nutrients are added periodically to the reactor, and the culture is only harvested at the very end of the fermentation. It is preferable to normal batch mode (where no media or nutrients are added during the course of the cell culture) because it leads to higher yields. Fed-batch fermen-

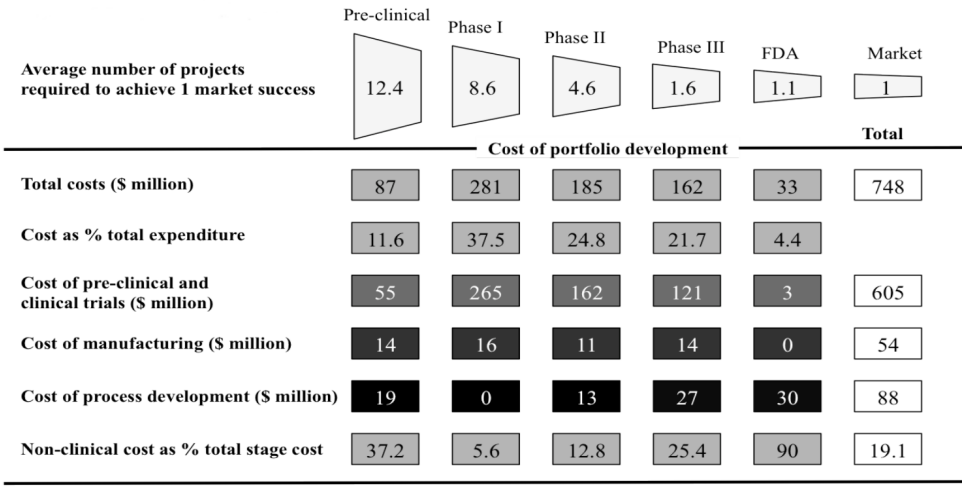


Figure 1.1: Drug development pathway and its associated costs (Nie, 2015)

tation is a well characterised process, which is perfectly suited towards products which can remain stable over the duration of the cell culture. In perfusion mode, media and nutrients are also continuously added to the reactor, but the product is harvested throughout the culture, rather than at the end in fed-batch mode. This mean that potentially harmful by-products or waste are continuously removed, thus becoming particularly useful when a product is not stable, where the residence time must be kept low. This in turn means that the cell density that can be achieved with perfusion is higher than that of fed-batch, thereby increasing productivity. Other advantages of perfusion mode over fed-batch mode include lower capital investment costs, due to the smaller reactors that are required, and the fewer number of seed train reactors that are necessary. Also, contamination is less of an issue with perfusion, since product that was harvested prior to the

Table 1.1: Development times of creating a new biological therapeutic (Nie, 2015)

		Phase			
Stage	Pre-clinical	I	II	III	FDA
Clinical trial duration (years)	1	1.6	2.4	2.7	1.5
Process development duration (years)	1	0	0.5	2	1.5
Manufacturing duration (weeks)	6	5	5	13	0

contamination is still viable (checks are made with every harvest, which is often daily), whereas with fed-batch mode the entire batch would have to be discarded. These advantages of perfusion mode over fed-batch mode can sometimes lead to manufacturers choosing perfusion during clinical trial phases (where production quantities are low and thus do not warrant the higher investment costs for batch systems), but then move to fed-batch mode for production quantity (Meuwly et al., 2006). The reason for this is that perfusion reactors are traditionally much smaller than fed-batch reactors, meaning that for large-scale production it is usually more efficient to use fed-batch reactors. Of course, the type of product being manufactured has a huge bearing on which process is chosen. Monoclonal antibodies are commonly manufactured using fed-batch fermentation, since they are relatively stable molecules, whereas blood factors such as Factor VIII would be too unstable to be manufactured under fed-batch mode, thus perfusion is used in these cases. Figure 1.2 shows the conceptual difference between perfusion and batch mode processes.

1.2 Problems Facing the Biopharmaceutical Industry

Risks involving clinical trial failure are obviously important parameters that need to be considered when developing new drugs, especially with the high costs involved, but there are also issues with the manufacturing. In recent years, around 27% of all new medicines in active development come from biopharmaceuticals, but many of the unit processes involved in the manufacture of these products are not fully characterised, creating fluctuations in the performance and productivity of the entire process (Ündey et al., 2010). Typically the process will involve fermentation followed by cell harvesting and product recovery, and finally purification and formulation. There are also the concerns of sterilisation, quality control and assurance, validation and regulatory approval to take into consideration. The European Medicines Agency (EMA) regards quality, safety and efficacy as the three main criteria upon which to approve new drug candidates (Benzi and Ceci, 1998). The fact that biological material can often be unpredictable is one

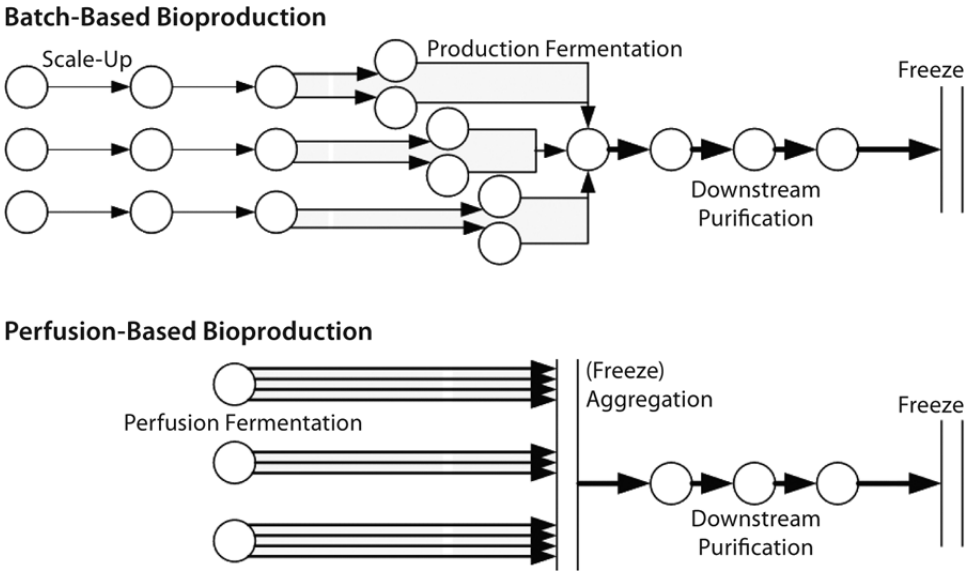


Figure 1.2: Schematic of batch and perfusion modes (Acuna et al., 2011). Typically, batch-based processes have downstream sized according to the size of one reactor. Since the culture duration can be quite long (>10 days), multiple vessels are staggered so that the downstream equipment is more efficiently utilised. Also, the larger vessel sizes in batch-mode require more seed train vessels for scale-up. In perfusion mode, material is harvested continuously and sometimes frozen, before being processed downstream. Manufacturers may choose to freeze the material before DSP to increase flexibility, thereby completely separating the USP from the DSP.

of the main challenges that biopharmaceutical companies face. The biological nature of the product manifests itself in other problematic areas, such as more stringent regulatory control, leading to extra costs being incurred in the purification stages of biomanufacturing, namely the chromatographic steps, which in turn increases the overall cost.

The basic hurdles that biopharmaceutical companies strive to overcome include reducing manufacturing costs and product development times, increasing manufacturing productivity, and ultimately increasing a product's profitability. Many of these hurdles are shared with the pharmaceutical (chemical) industry, but owing to the factors outlined above biopharmaceuticals are under larger pressure. Another key issue that pharmaceutical companies are facing is that of patents expiring. For example, analysts in 2011 estimated that Eli Lilly could see a 50% reduction in sales by 2020 as their main drugs come off patent and lose exclusivity (Edwards, 2011). AstraZenica and Pfizer both face similar outlooks,

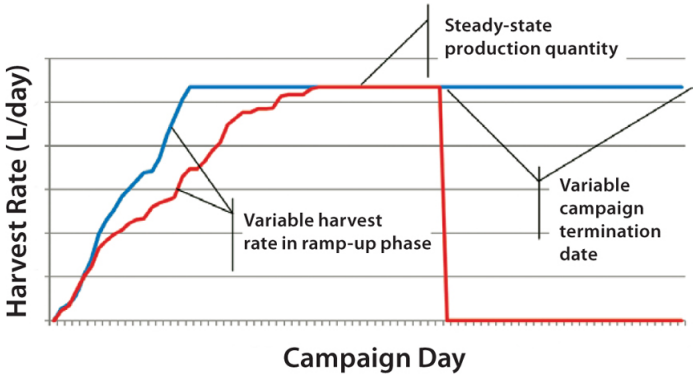
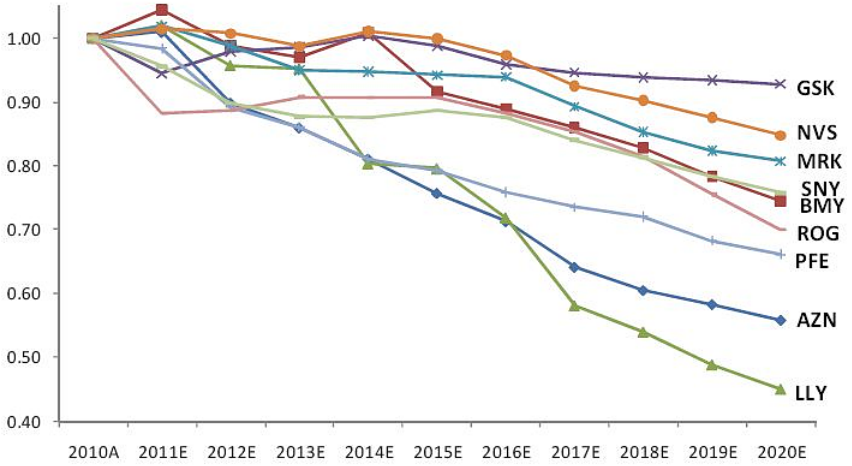


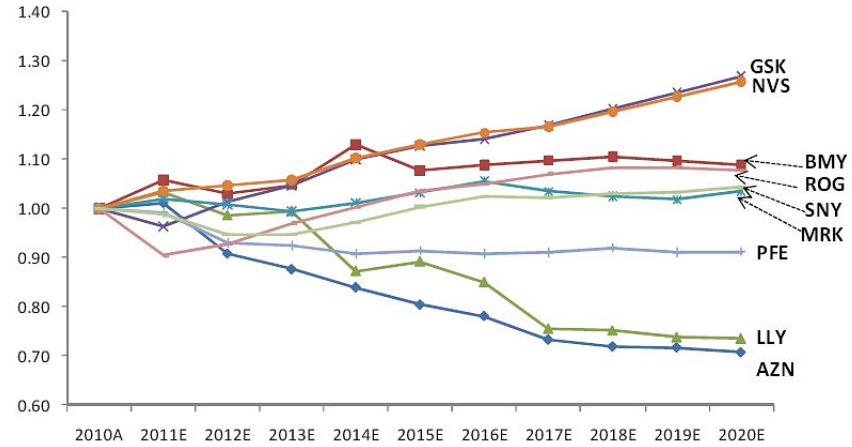
Figure 1.3: Diagram showing how a perfusion process changes over time (adapted from Acuna et al. (2011)). During the ramp-up stage, cells are growing, and thus the harvest rate is gradually increased until it reaches a steady state. Sometimes manufacturers discard material harvested during the ramp-up stage. The last stage is termination, which is not necessarily fixed, but is often preferred to be kept constant. The graph shows two processes with different termination points.

as shown in Figure 1.4. Such patent expirations have been forcing companies to consider either acquiring another smaller biotechnology company which has drugs in the pipeline, or be acquired itself. Pfizer's CEO has discussed strategies involving breaking the company into smaller parts, leading to a smaller but more profitable drug company (Barry, 2015). Pfizer has also cut costs by slashing jobs and closing down facilities in R&D, saving approximately $1.5bn (Inman and Hawkes, 2011). Other ways in which to cut costs are being investigated by biopharmaceutical companies, so that they remain competitive even after patents have expired. Decision-making frameworks which optimise portfolio selection and capacity planning are examples of areas which are currently being researched, and are the premise of this piece of work.

There are various examples of the repercussions of incorrect capacity planning, including high profile company acquisitions owing to over- and under-capacity respectively (Ransohoff, 2004). Capacity sourcing strategies for biopharmaceutical companies often involve consideration of build-versus-buy decisions, i.e. choosing whether to outsource manufacturing to a contract manufacturing organisation (CMO) or build in-house facilities (Langer, 2011). Developing a comprehensive production planning strategy requires careful assessment of the cost, risk, and time trade-offs of each option (George et al., 2007).



(a) Base pharmaceutical revenues to 2020, normalised to 2010



(b) Total company revenues (Base + Pipeline + Non-Pharma divisions) to 2020, normalised to 2010

Figure 1.4: Comparison between major pharmaceutical companies' future revenue estimates (Edwards, 2011). Companies include GlaxoSmithKline (GSK), Novartis (NVS), Merck (MRK), Sanofi (SNY), Bristol Myers Squibb (BMY), Roche (ROG), Pfizer (PFE), AstraZenica (AZN) and Eli Lilly (LLY).

Decisions to build a facility for commercial production need to be scheduled several years in advance before a drug's full market potential, likely dose range, cell line productivity and process yields are known. The use of CMOs enables such capital outlays to be delayed whilst incurring a premium for their services. A further factor affecting the decision relates to the relative difference in manufacturing efficiencies assumed between in-house and external manufacturing. In the case study presented in this paper, third party manufacturers were assumed to have higher manufacturing yields than the drug developer company (Lakshmikanthan, 2007).

By outsourcing to CMOs, biopharmaceutical companies can mitigate risks concerning failed batches, natural disasters, incorrect market demand forecasts, or a clinical trial failure. The downside of using CMOs is usually the loss of process control, or delays in technology transfer to in-house facilities if later required (Blackwell et al., 2010). Building a new facility on the other hand, requires consideration of the lead time for construction, commissioning and validation of the facility, all of which can take up to four years to complete, and can cost $40-650M for large commercial antibody facilities (Farid, 2007).

1.3 Current Industrial Practice

Presently, there are no software packages which conduct true biopharmaceutical capacity planning within an optimisation framework. Currently, production plans are created manually in an Excel spreadsheet or by using Microsoft Project (or its equivalents). For small numbers of products/facilities this is a feasible strategy (albeit non optimal). As portfolios increase in size, automated methods need to be devised, hence the purpose of this work.

There are simulation-based programs which are used for biopharmaceutical manufacturing. BioSolve Process (Biopharm Services, Chesham, UK) is an Excel-based software package which allows a user to create bioprocesses from a set of predetermined unit processes, and then calculates the costs that the user would likely observe given a certain annual throughput. A limited amount of scheduling

can be conducted, but the software is not designed to be used for scheduling of a multi-product facility, nor is it designed to be used from a higher level capacity planning perspective. The scheduling information it provides is in hours, but for one batch only. This is not particularly useful when a 10 year plan is being considered. It is, however, a good piece of software when a process engineer wants to tweak a process in order to achieve higher yields or when trying to minimise the size of unit operations. In fact, BioSolve can be used as an input for many of the parameters (costs and yields) in the model presented in this thesis.

ExtendSim (ImagineThat! Inc, San Jose, USA) is a discrete-event simulation package which has its own programming language enabling users to create their own processes. This dynamic modelling framework, which allows a user to explicitly define the mass balance equations in each unit operation, offers more flexibility at the expense of increased complexity. Whilst it can be an excellent tool to use at a process level, it is not widely used for capacity planning. It has been used to evaluate the operational, economic and environmental characteristics of fed-batch and perfusion bioprocesses (Pollock et al., 2013). ExtendSim was also used to conduct capacity planning for one facility over a time horizon of one year (Ashouri, 2011). Brute force was used over a selection of campaign combinations in order to find an optimal schedule. This was made possible by the fact the model was very small, and only 12 different manufacturing scenarios were considered.

INOSIM (INOSIM Software GmbH, Dortmund, Germany) is another package which allows a user to design a process and conduct mass balancing. It can also carry out optimisation at the process level, covering production costs, tank dimensions, sequences of units, or other parameters. These optimisations are at the process level, and therefore cannot be regarded as capacity planning for multi-product multi-suite biomanufacturing. The three software packages mentioned in this section each have their merits, but are not the right tools for long-term capacity planning. Hence, new techniques must be investigated for this purpose.

1.4 Mathematical Programming

During World War II, the allied forces were under huge amounts of pressure to supply ground troops with food and weaponry in the shortest amount of time and at the lowest transportation cost. Similar issues were raised when tasked with finding the optimum pathway for destroying German U-boats, so that the British and American fleets would not be destroyed whilst delivering supplies. Great mathematicians were recruited into solving these optimisation problems, and from these studies came the birth of the renowned simplex method. Mathematical programming as it is known today owes its initial development to this time period, but has since then evolved through the use of more advanced techniques. In this section a discussion will be made on how mathematical programming has been used in the biopharmaceutical industry, and how some of the mathematical techniques used have progressed over the decades.

1.4.1 Linear programming

Linear programming is a branch of mathematical programming which derives its name from the fact that the mathematical expressions used in the constraints and objective function are all linear. *Programming* is a slight misnomer in that it does not refer to any computer programming, but rather the older definition of the word meaning 'planning'. One may think that modelling the world using linear equations is not particularly useful, since many problems that occur in practice do not exhibit linear relationships, and would therefore be inaccurately modelled. In reality however, linear programming has been shown to be useful in many cases, especially those which involve scheduling (Lorigeon et al., 2002), capacity planning (Papageorgiou et al., 2001), transportation (Abara, 1989) and distribution (Eraslan and Derya, 2010). In 1970 IBM stated that approximately 25% of all scientific computation was dedicated to linear programming (Chinneck, 2000). Although its application to the biopharmaceutical industry is not as prevalent as to that of the chemical industry (partly due to the added complexities of modelling biological processes), there have been developments in recent

years owing to biopharmaceutical companies wishing to seek alternative methods of cost-cutting. Research conducted on the pharmaceutical, food and beverage, and certain specialised chemical industries (Moreno and Montagna, 2009; Lázaro et al., 1989) can be applied to the biopharmaceutical industry since they commonly deal in terms of batches. The number of batches must be an integer number, and thus techniques used to solve mixed integer linear programs (MILP) have been developed (see Section 1.4.2).

Early research into capacity planning has been reviewed by Papageorgiou and Pantelides (1996), in addition to which a general mathematical formulation for multiple campaigns in multi-purpose batch plants is also presented. Descriptions of how particular characteristics of campaign-based batch processes, such as campaign changeovers and inventory profiles, were addressed. A mathematical MILP formulation encompassing strategies for product development, capacity planning and investment for pharmaceutical industries has been created (Papageorgiou et al., 2001). They outlined the various characteristics present in modelling the pharmaceutical industry as well as the significance of taxation, different sales regions and other financial attributes in obtaining a meaningful solution. They also mentioned how scale-up and qualification constraints could be used to model the extra time and cost required to start manufacturing a product in a facility for the first time. It should be noted that the model used time periods of one year, and hence it was solely to be used for capacity planning rather than scheduling.

Biological systems often show great variability in productivity during early development, and thus attempts to capture uncertainties within the model are important. Capacity planning for three products under uncertainty in clinical trials has been addressed by Rotstein et al. (1999), where the model was used to determine whether plans for the investment into future manufacturing capabilities should be made. Gatica et al. (2003a) build upon these models, but instead of clinical trials being either a success or failure as seen in other research conducted (Maravelias and Grossmann, 2001), they created four levels of product success, resulting in 4^N scenarios in the final stage of the model (with N being the number of products). Overall it became a large scale stochastic programming problem,

which in their case of four products was not too problematic to solve, but the problem could easily escalate in complexity. Thus Gatica et al. (2003b) discussed using a scenario-based aggregation/disaggregation procedure to provide a more efficient solution strategy without compromising the quality of the final solution. The MILP model that was formulated was used in assisting the product portfolio and investment decision-making. A framework which includes both stochastic simulation and an MILP model was described by Varma et al. (2008). They created an integrated resource management tool with the goals of maximizing a pharmaceutical portfolio's expected net present value (ENPV), controlling risk and reducing drug development cycle times.

Lakhdar et al. (2005) developed a mathematical formulation for the planning and scheduling of a multi-product biopharmaceutical manufacturing facility, and showed it to be more efficient in terms of facility utilisation and cost reduction than the standard industrial rule based approach. This model, which was formulated as MILP, was later expanded into a multi-facility and multi-product model whereby fluctuations in demand were considered, as well as multi objective criteria such as customer service level and facility utilisation by means of goal programming (Lakhdar et al., 2007). Lakhdar and Papageorgiou (2008) also illustrated how a different optimisation algorithm could be used to provide greater optimisation over deterministic approaches when carrying out Monte Carlo simulations on uncertain fermentation titres. Sousa et al. (2008) discussed a multi-stage approach being applied to an agrochemical industrial case study (but also applicable to pharmaceutical cases) whereby initially in the first stage the production and distribution plan is optimised for a one year time horizon, and then the results from this stage are fed into the second stage where a detailed schedule with a smaller time horizon is calculated. A new technique for the calculation of production profiles for large multi-product facilities was shown by Sung and Maravelias (2006), where an offline analysis of the MILP problem allowed for linear constraints to be added to the model, producing high quality scheduling information and solutions, without being as computationally expensive.

Short-term scheduling of batch plants with sequence-dependent changeover

times has been addressed using continuous-time representation MILP models with either binary variables or extra constraints (Castro et al., 2006). Combined planning and scheduling models can be computationally expensive and have been tackled by different approaches such as the multi-stage MILP approach described by Sousa et al. (2008), and by mathematical programming formulations with separate scheduling and planning aspects of supply chain optimisation which are then linked sequentially via a common time basis (Amaro and Barbosa-Póvoa, 2008).

1.4.2 Techniques

Simplex method

Early methods of linear programming (Kantorovich, 1960) were later refined by the simplex method (Dantzig, 1951). The techniques used in most commercial linear programming solvers are based around this simplex method, the details of which are beyond the scope of this review. There are a few important points to make though, so that one can understand why the simplex method does not fair so well in certain situations. Figure 1.5 shows how the simplex method moves

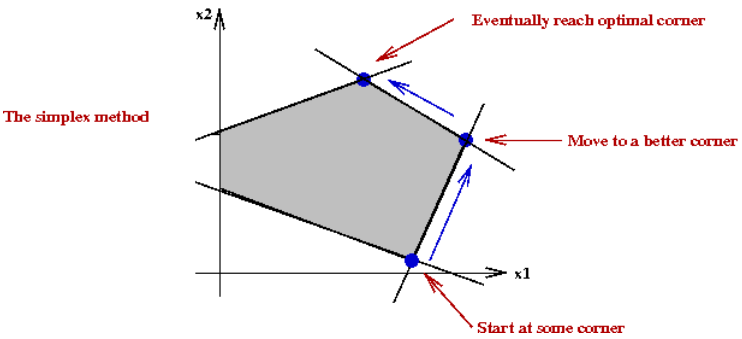


Figure 1.5: Graphical illustration of simplex method with five constraints. The feasible region is shaded in grey.

from one corner of the feasible region to another until there are no better adjacent corners, at which point it has found the optimum value for the objective function. The corners are always optimal (i.e. no point lying on a constraint line will ever be better), so only corners need to be checked. To solve the problem algebraically though, a simplex tableau is formed and slack variables are added to inequality

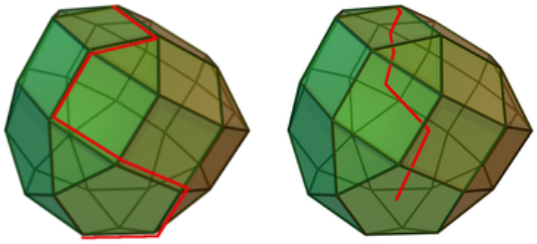
constraints as shown below:

$$x_1 \geq 0 \rightarrow x_1 + s_1 = 0$$

Although more variables are introduced (by means of adding slack variables, and artificial variables in more complicated examples), this does not turn out to have a large effect on the time it takes to solve the problem, since the speed of the solution depends largely on the number of constraints. Adding more constraints creates further corner points, and it is these corner points which need to be traversed which consumes the most time (moving to adjacent corners involves costly pivot operations in the tableau). The algorithm runs very efficiently in practice, generally in $2m$ to $3m$ iterations, where m is the number of constraints (Zadeh, 2008). Although the average cases run in polynomial time, in some pathological cases the solution complexity can become *exponential* (Klee and Minty, 1972). Some problems which are very large with many constraints perform badly with the simplex method, hence alternative techniques were developed.

Interior point method

The simplex method involves moving from one corner point to another, and thus it will always lie on the surface of the polyhedron (shaded feasible region). The interior point method (Karmarkar, 1984) allows for movement into the polyhedron, as shown in Figure 1.6, and thus can provide a more efficient way of reaching the optimum for very large problems. The commercially sold CPLEX solver includes variants of this algorithm rather than using just the standard simplex method (Darby-Dowman and Wilson, 2002). It has been shown that while large problems are solved quicker via the interior point method, small to medium sized problems are still better suited towards the simplex method (Paparrizos et al., 2003). This is down to the fact that while the interior point method can quickly get close to the optimum by skipping through corner points, it then takes a long time to truly reach the optimum.



(a) Simplex method (b) Interior point method

Figure 1.6: Comparison between solution pathways for simplex and interior point methods.

Dual simplex method

The dual simplex method uses the interesting relationship of the mirror-image of the linear model to reduce the solving time. Every model (primal) has a mirror-image (dual) which can be thought of as the tableau configured sideways. If the primal model has more constraints than variables, then the dual model will be the opposite way round - fewer constraints and more variables. As mentioned previously, problems are quicker to solve when there are fewer constraints, hence the dual simplex method can prove to be very beneficial for large problem sets. There have been studies carried out showing that a 94-fold reduction in time over the standard simplex algorithm can be achieved via a primal dual algorithm (Paparrizos et al., 2003). It has the additional benefit of being able to be used in conjunction with interior point methods, again improving performance.

Branch and cut algorithm

The simplex method and interior point method can be used to solve problems with continuous variables, but are not able to cope with discrete variables. Discrete variables are those which cannot take real values, for example integer variables cannot have fractional values like 1.5. This makes the problem much harder to solve, and forms a new branch of mathematical programming called Mixed Integer Programming (MIP). One way of solving these problems is by enumerating every possible solution and then picking the best one, but this would be very unwieldy for large problems. Thus a technique using the branch and cut algorithm is used

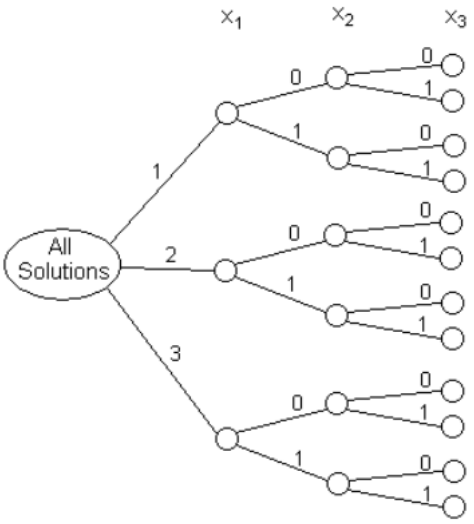


Figure 1.7: Tree-like structure of the enumeration of a problem (Chinneck, 2000)

to reduce the number of enumerations that are required. The branch and cut algorithm is a hybrid between a branch and bound algorithm and a cutting plane method. Consider a problem which has one integer variable ranging between 1 and 3, and two binary variables. Figure 1.7 shows the enumeration of all possible solutions, which in this problem is small, but can easily be much larger. The branch and bound technique grows the tree in stages, so that fewer nodes need to be visited. For example, say there was a constraint $x_1 - x_2 - x_3 \geq 0$ for the previously described problem, then it is very clear that x_1 can never equal 1, and thus all the nodes on this branch at removed (and thus the search space is reduced). The algorithm calculates a bound on the best value that can be achieved by the objective function if the tree were to be expanded further, and will only expand it if the value is greater than what it currently has (for a maximisation problem). This bound is an estimation of the *best* case scenario, since it is actually a relaxation of the original problem. So, the branch and bound technique effectively splits up the variables into their discrete values, and only expands the tree if the estimated bound is greater than what has currently been achieved.

The cutting plane method does not split up the variables, but instead adds a cutting plane into the problem. For example, if the constraint $x_1 + 3x_2 \leq 5$ did

not produce an integer feasible solution, the cutting plane method may choose to change the constraint to $x_1 + 3x_2 \leq 4$ and see if an integer solution is obtained this time. A combination of both branching and cutting is what makes the B&C algorithm powerful, and allows very large problems to be solved more easily. There are many intricacies in the B&C algorithm, and it is up to the developer of the algorithm to make it more efficient. The mathematical modeller should be aware of how it works, but should not necessarily need to delve into the details of its implementation. Colvin and Maravelias (2010) describe how they developed a novel branch and cut algorithm which can reduce the time required to obtain an optimal solution. It was applied to scheduling of clinical trials in pharmaceutical research, and they illustrated that by understanding the real-world problem they were able to adapt the algorithm to remove nodes from the tree that were unnecessary, thus increasing speed. They mention that although the methods were specific to a particular case, they could also be applicable to a general class of problems.

1.4.3 Multi-objective methods

Most work involving capacity planning revolves around optimising single-objective models. Usually the objective under consideration is total cost or net present value (NPV). However, models which can incorporate multiple criteria are better placed to provide more holistic manufacturing schedules which meet the various conflicting objectives a biopharmaceutical company may have.

In terms of the stage at which a decision maker makes his/her preference, there are three categories of multi-objective optimisation: the *a priori* methods, the interactive methods and the *a posteriori* or generation methods (Hwang and Masud, 1979). An example of *a priori* methods would include weighted-sum goal programming, whereby a decision maker makes a preference prior to optimisation by setting goals and weights in the objective function. The main issue with this type of method is that it is difficult to determine beforehand which goal targets and weights should be used. In the interactive methods, a decision maker reaches the most preferred solution through dialogue with the multi-objective model. The

search process will eventually converge to a solution that is most suitable given the responses by the decision maker. However, this method prevents the user from being able to see the entire decision space. In the *a posteriori* methods, a complete set of efficient solutions is generated, and then the decision maker selects the most suitable solution given his/her criteria.

There is extensive literature surrounding multi-objective optimisation of supply chain management. Amodeo et al. (2007) developed a simulation-based multi-objective optimisation method for the inventory policies of supply chains. They showed that their approach was able to obtain better solutions in terms of two objectives: total inventory cost and service level. Roghanian et al. (2007) considered a probabilistic bi-level linear multi-objective programming problem and applied fuzzy programming techniques adapted from Osman et al. (2004) to deal with uncertain input parameters. As previously mentioned, Lakhdar et al. (2007) incorporated multiple objectives, including cost, customer service level and capacity utilisation, into a biopharmaceutical capacity planning model via the use of goal programming. Vahdani et al. (2012) developed a bi-objective mathematical programming formulation which minimizes the total costs and the expected transportation costs after failure of facilities in a logistics network.

The ϵ-constraint method is an a posteriori method for multi-objective optimisation, and has been used in the context of supply chain management. Bashiri et al. (2014) describe its use in a supply chain network for the objectives of cost and customer satisfaction. The ϵ-constraint method was also used to generate Pareto-optimal curves in a bi-criterion non-convex MINLP for the global optimisation of chemical supply chains (Guillén and Grossmann, 2010). Pishvaee and Razmi (2012) used the ϵ-constraint method to consider multiple environmental impacts beside the traditional cost minimisation objective. Pozo et al. (2012) use principal component analysis to reduce the number of objectives that need to be considered within a chemical supply chain, and then use the ϵ-constraint method to generate a set of Pareto solutions. Guillén et al. (2005) combined the ϵ-constraint method with a two stage programming model to tackle the problem of design and retrofit of a supply-chain network consisting of several production

plants, warehouses, and markets, and the associated distribution systems. The objectives considered were NPV, demand satisfaction and financial risk, with a set Pareto of solutions generated to aid the decision maker. Mavrotas (2009) presented a novel version of the ϵ-constraint method which avoided the generation of weakly Pareto optimal solutions and increases performance by removing redundant iterations. The authors then improved the method with particular attention to multi-objective integer problems (Mavrotas and Florios, 2013).

1.5 Alternative Heuristic Search Methods

Although formulating the problem using mathematical modelling allows for the use of high performance solvers, sometimes the problem is too large to be solved in reasonable time, and other times the problem is too complex to be described as linear. In these cases, heuristic search methods can provide alternative methods of arriving to an optimised solution. They may not be mathematically the best solutions, but they can be very close to the optimal value, and the added benefit of being able to model more complex situations with greater flexibility can outweigh the downsides.

1.5.1 Simulated annealing

Simulated annealing is one of the older heuristic search methods (Metropolis et al., 1953), and has been used for a variety of problems. Its name comes from annealing in metals, whereby the metal is heated and then cooled down slowly, thus increasing the size of its crystals and reducing their defects. The heat gives the atoms energy to move away from their original positions (which can be classified as a local minimum of the internal energy) and move randomly through states of higher energy; the slow cooling gives them more chance of finding configurations with lower internal energy than the initial one. In combinatorial optimisation, it works by searching through the entire problem space, preventing itself from becoming trapped in a local optimum by allowing itself to move to inferior solutions under certain conditions. Switching to an inferior solution is

dependent on an acceptance probability function, which takes into account the change in solution value (Δc), and temperature (T):

$$P() = \begin{cases} 1 & \text{if } \Delta c > 0 \\ e^{-\frac{\Delta c}{T}} & \text{if } \Delta c < 0 \end{cases} \tag{1.1}$$

If P() is less than a uniform random number, $R \in [0, 1]$, then a move to the newly calculated solution will take place. Thus, if a solution is inferior to the previously calculated solution, the algorithm may still change to it depending on the probabilistic outcome of Equation 1.1. The temperature is reduced after each iteration ($T \leftarrow \alpha T$, where α is a constant close to 1), thus reducing the chance of switching to an inferior solution as the iteration process goes on. The initial temperature that is used is important, as this will determine how easily it switches to inferior solutions at the beginning - starting with a low temperature may result in becoming trapped in a local optima very quickly. Choosing an initial temperature requires some knowledge of the problem, and can take trial an error to get right. It should be noted that the number of iterations is dependent on the initial temperature used, the α constant used to reduce the temperature, and the final temperature (the temperature at which the process is stopped). The final temperature is again somewhat problem dependent, but Lundy and Mees proposed stopping when:

$$T \leq \frac{\epsilon}{\ln[(|S| - 1)/\theta]} \tag{1.2}$$

where S is the solution space, and the final solution is within ϵ of the optimum with probability θ (Lundy and Mees, 1986).

Ku and Karimi (1991) showed one of the first applications of simulated annealing in scheduling problems, and reported that out of the four algorithms that they tried using, the simulated annealing algorithm provided the best solution, although at the expense of greater CPU time when compared to their other iterative algorithms. A similar result was obtained by Tandon et al. (1995), where they showed that a simulated annealing algorithm provided better solutions than

those given by other heuristic methods and the list scheduling algorithm. Their approach incorporated sequence dependent clean-up times, and they measured performance based on tardiness minimisation (i.e. ensuring products are delivered on time). There are examples of simulated annealing being used in computer science, one being capacity planning of networks (Habib and Marimuthu, 2010). They showed how the use of SA algorithms allowed them to cut network traffic by 20%, thereby increasing their overall network capacity and reducing maintenance costs. Other work (Tsenov, 2006) showed how SA algorithms could be used to optimise telecommunication networks by using different criteria such as network reliability, restricting traffic congestion below a certain threshold, and respecting a maximum transit time (the time for which a packet of information is travelling through the network). In terms of biopharmaceutical manufacturing, this could be interpreted as backlog delays, facility utilisation, and product shelf-life respectively. Another example of SA being used is for the optimisation of investment in a transportation network under uncertainty (Sun and Turnquist, 2007). The model sought to maximise expected system capacity, subject to uncertainty of future demand, and this had the effect of the model finding investment plans that will create capacity flexibility as well as increasing expected capacity.

1.5.2 Genetic algorithms

Genetic algorithms (GA) are part of a branch of meta-heuristics called evolutionary algorithms, termed as such because they derive much of their operating characteristics from situations arising in biology. The technique for genetic algorithms (Goldberg, 1989) starts by using a collection of solutions (known as a population of chromosomes), and then using selective breeding and recombination strategies, better solutions are produced. Generally, the optimisation stops when a certain number of generations have been produced, or when a satisfactory solution has been reached. In terms of recombination, different numbers of crossovers between chromosomes can be used to vary the offspring, and the mutation rate can be varied too. Some studies have shown that having both a higher mutation rate and different rates for different bits on the chromosomes

can be beneficial; in fact it may also be useful to increase the mutation rate as the search progresses.

Berning et al. (2004) have shown how genetic algorithms can be used for supply chain optimisation in the chemical process industry. They describe how the scheduling can be split up into two distinct parts: long-term planning which look far ahead and provides a rough sketch of production capacity, and short-term scheduling which considers production sequencing, keeping idle time and inventory low, and all the other production constraints that are present. They mention how mathematical modelling is often not the most ideal tool to use, since many production constraints such as sequence dependencies and lot size restrictions lead to NP-hard optimisation problems (Monma and Potts, 1989).

Recently, Ramteke and Srinivasan (2011) showed how GAs could be integrated with a graph-based network structure so as to speed up the solution time. The optimisation was concerning large-scale refinery crude-oil scheduling, where the problem involved multiple objectives. They showed a significant reduction in CPU time when compared to a standard MILP formulation, from 2988 seconds to 34 seconds, with only a small decrease in profit for the GAs. Urselmann et al. (2009) described the use of a hybrid algorithm, which they term a 'memetic algorithm', which incorporates both GAs and local mathematical solvers. The combination of the two optimisation methods reduced the overall search space, and allowed for large global optimisation. The memetic algorithm exploits GAs' ability to escape local optima, and uses a local NLP solver to optimise large continuous problems locally. Together, these two methods gave a 75% increase in speed in certain conditions when compared to an alternative algorithm called OQNLP. This alternative algorithm is a scatter search based multi-start heuristic, and works by generating multiple starting points from which a local NLP solver (CONOPT in this case) starts its optimisation.

Estimation of Distribution Algorithms (EDAs), which are a branch of GAs, have been used in a multi-objective optimisation framework, where the three main criteria that were addressed were portfolio management, scheduling of drug development and manufacturing, and whether or not third parties should be used

for manufacturing or development of candidate drugs (George and Farid, 2008a). The model built upon previous work (George et al., 2007) where simulation was used in a multi-criteria decision-making framework to aid companies when faced with the acquisition of commercial-scale biopharmaceutical manufacturing capacity. The detailed economic model from this work, alongside with the genetic algorithms added in the optimisation framework, allowed George and Farid to show that by taking multiple drug candidates into consideration rather than just one single drug, the overall risk to NPV can be reduced, although one of the side effects of reducing NPV risk is that the overall mean NPV is reduced. The results suggested the integration of all activities in-house in scenarios without budgetary constraints. However, in scenarios with budgetary constraints, the results indicated that managing risk through outsourcing to CMOs and sharing capacity with partners would be a more optimal strategy. Hence, the optimization outputs propose committing to creating capacity as late as possible with limited budgetary constraints. However, the key point of the work was that an optimisation framework, using evolutionary algorithms and machine learning, had been used to solve portfolio development and capacity planning simultaneously, something which had not been done before.

1.5.3 Swarm intelligence

Swarm intelligence is a branch of artificial intelligence which takes ideas from behaviour prevalent amongst social insects or animal societies, and applies them to the design of multi-agent systems. Techniques using swarm intelligence for optimisation have recently become popular, largely due to their ability to deal with complex problems in a robust and flexible manner. Two of the more successful techniques are ant colony optimisation and particle swarm optimisation, the first of which will be discussed here. The use of ant colony optimisation (ACO) in combinatorial optimisation was first described by Dorigo et al. (1991), the inspiration of which came from observing how ants forage for food. Figure 1.8 shows a summary of how ant foraging works.

A good explanation of how ant foraging can be applied to optimisation prob-

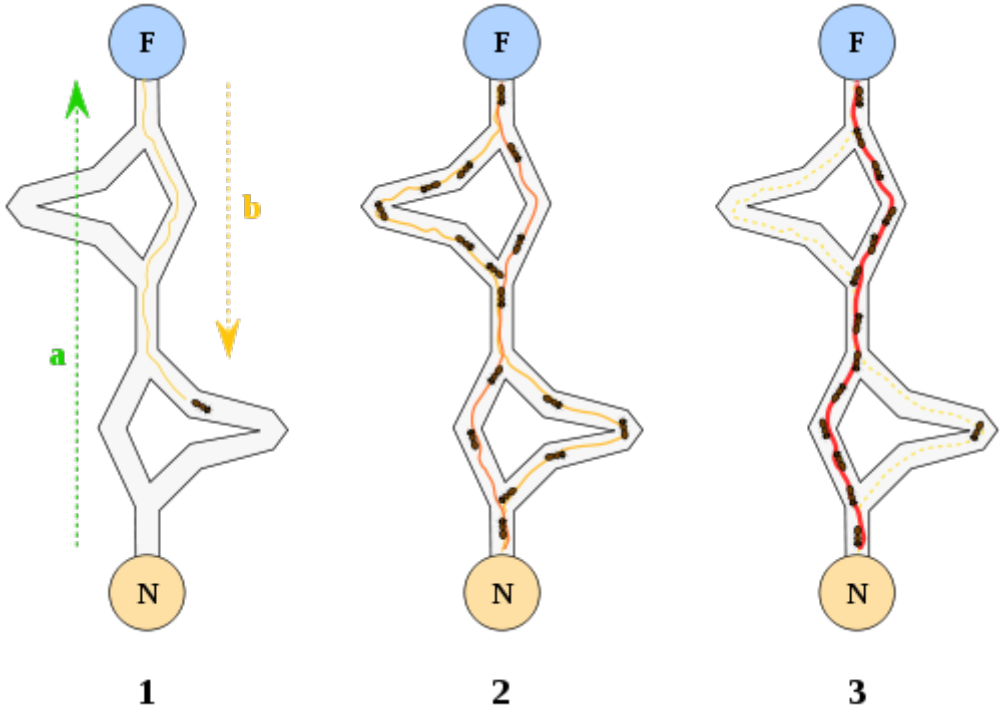


Figure 1.8: Shortest path find by an ant colony (Dréo, 2006). Ants can follow any of the four routes from the nest (N) to the food source (F). As they return to the nest, they lay a pheromone trail. The ant which took the shortest route will return first, and thus the probability of the shorter path having more pheromone (which influences the ants' decision on which path to take) will be higher. The net effect is that over time, almost all the ants will follow the shorter path.

lems can be found in paper by Blum and Li (2008), where they outline a framework that can be used to solve the travelling salesman problem (TSP). Compared to other state-of-the-art techniques, the original ACO algorithm was not as good at solving the TSP, and thus different variants of the ACO framework came into existence, mainly varying in the rules applied to pheromone update (Dorigo, 1992; Dorigo and Gambardella, 1997; Stützle and Hoos, 2000). ACO has been applied to a number of problem types, including bioinformatics (Shmygelska et al., 2002), scheduling (Merkle et al., 2000), multi-objective problems (Guntsch and Middendorf, 2003), and dynamic problems (Guntsch and Middendorf, 2001). One of the problems with ACO is that when a problem becomes highly constrained (for example, in scheduling problems), ACO performance is inferior to other methods of optimisation. This is also seen with other search heuristics, the reason being that when a problem in not excessively constrained, the hard part becomes

optimisation rather than finding a feasible solution. In these instances, ACO algorithms and other meta-heuristic algorithms perform well. However, when the problem is very constrained, the difficulty lies in finding feasible solutions rather than the optimisation. Restricting the search space to promising regions is part of something called constraint programming, and has been hybridised with ACO to enable its use to more challenging problems (Meyer and Ernst, 2004). Wang and Chen (2009) developed an ant algorithm that can solve non-linear mixed integer programming models which maximise profit through capacity planning and resource allocation. They used constraint programming techniques mentioned previously to deal with the problem's inherent complexities, and found that the solutions provided by the algorithm were equal to that of genetic algorithms.

1.6 Justification of Mathematical Programming Approach

This work focuses on MILP methods to determine optimal manufacturing schedules. Other methods have been highlighted in this literature review, but none provide the proof that a solution is globally optimal. Furthermore, biomanufacturing capacity plans have not been researched extensively using heuristic methods, whereas encouraging attempts have already been made in mathematical programming. Whilst literature for mathematical techniques in biomanufacturing are limited, there is extensive research that has been conducted in other sectors for the case of capacity planning. Other techniques can also be investigated in tandem, but they should ultimately be compared to exact methods, hence this thesis predominately focuses on MILP methods.

1.7 Aims and Organisation of Thesis

The previous sections have described the main issues the biopharmaceutical industry are currently facing, and how these pressures influence decisions regarding capacity planning. Optimisation techniques addressing how capacity planning

challenges have been solved in other industries have been discussed. Mathematical techniques such as mixed integer linear programming and alternative search heuristics such as genetic algorithms have been investigated in the context of biopharmaceutical capacity planning. Despite the attention that has been given to this problem domain in literature, cases where both perfusion and fed-batch processes are present have not been considered. As manufacturers start to see the benefits of using perfusion systems, there will be a greater need for optimisation models that can cater for these processes.

The aim of this thesis is to develop a computational decision tool which can provide biomanfacturing production plans for different modes of cell culture. In particular, it should provide:

- Modelling of perfusion mode and fed-batch mode cell cultures
- Manufacturing schedules for long-term planning horizons
- Biomanfacturing costs and capital investment profiles
- Optimal selection of capacity expansion options
- Analysis and optimisation surrounding multi-criteria strategic decision making
- Analysis of the impact of uncertainty on biopharmaceutical capacity planning

The aim of this thesis is therefore to create a framework that produces optimal solutions to biopharmaceutical capacity planning problems, considering various capacity expansion options and different product types. The remainder of this thesis is structured around achieving these aims.

Chapter 2 discusses the problem domain in greater detail, including the model's input requirements and expected outputs. The need for an automated decisional tool is highlighted by an illustration of the computational complexity of the problem. Finally, an overview of how the framework is constructed is presented.

Chapter 3 outlines the challenges present in biopharmaceutical manufacturing when both perfusion and fed-batch processes must be considered. A discrete-time mixed integer linear program is created which accurately models both perfusion and fed-batch processes to produce optimal capacity plans. Sequence-dependent changeovers are introduced to correctly model the increased time required to switch between different process modes. To improve computational performance, a rolling time horizon is implemented.

The performance of the mathematical model is improved further in Chapter 4. A state task network (STN) representation is used to reduce the number of constraints and variables in the model, and improve computational efficiency and solution quality. The performance of the STN model is tested on two industrial case studies. New features are also added to the model, to further increase realism of the manufacturing schedules.

Chapter 5 discusses the addition of a multi-objective component to the STN model. Two methods are compared, weighted-sum goal programming and the ϵ-constraint method. The multi-criteria nature of biopharmaceutical capacity planning is explored via the consideration of various strategic objectives. An analysis of the impact these considerations can have on manufacturing schedules, capital expenditure and risk is discussed.

Chapter 6 outlines the important conclusions of this work, and possible avenues of extending the framework. Finally, Appendix B lists papers of the author published during the course of this work.

Chapter 2

Requirements and Analysis

This section will outline the problem being solved in more detail. It will discuss why the problem needs to be addressed, what information is required in order to solve the problem, and what exactly should be expected from the decision-support tool being developed. Finally, the components of the framework and how they work together are discussed.

2.1 Detailed Problem Statement

In order to reduce costs, biopharmaceutical manufacturers would like more guidance and assistance in decision-making regarding strategic planning. In terms of capacity planning, they would like to know when and where they should manufacture a product. This is simple for cases involving a small number of products in their portfolios, with one or two manufacturing facilities to choose from. However, as the number of products and facilities increases, so does the complexity of the problem, becoming much more difficult to solve manually. In order to better understand the problem, it is first necessary to discuss some of the constraints and inputs which influence the decision-making.

First of all, a list of the products and facilities that are to be included in the model need to be analysed. Different products will have distinct modes of cell culture (for example, fed-batch or perfusion mode). Facilities will also have their own capabilities regarding which products they can manufacture (see Figure 2.1).

Secondly, some information regarding the process needs to be ascertained. For a more detailed and complex model, information on the individual unit processes would be needed, but if the model is to assume a black box approach, then just the overall output information is required. For example, this could include data akin to the output of each batch (in kilograms), the time it takes to produce one batch, and the cost of manufacturing each batch. Then, using demand targets, one can begin working out which product needs to be produced where.

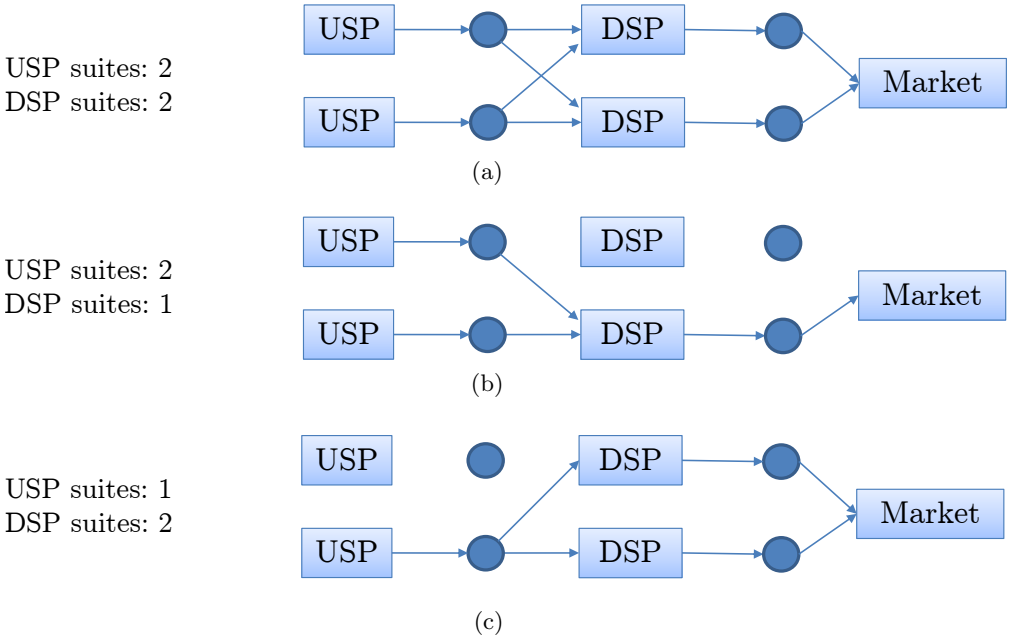


Figure 2.1: Capability matrices for a network of two USP suites and two DSP suites. The number of suites available for use for a particular product is shown on the left. In (a) both USP and DSP suites are available, in (b) only one DSP suite can be used, and in (c) only one USP suite is available.

The problem becomes more complicated when other factors are considered, such as product shelf-life (the product cannot be stored indefinitely but must be sold before it expires), an individual facility's storage capacity, and sequence dependent changeover times (Figure 2.2). The time required to switch between products includes the time to clean the suite and also move any equipment, and thus will depend on the equipment the processes use. Therefore, sequence dependent changeover times are required when the model contains vastly different product types. There are also options to manufacture in a CMO, or build a future

facility to cope with future demand. In fact, some facilities can also be retrofitted so that they are able to manufacture other products, again making the problem more complex. Figure 2.3 shows an overall view on some of the aspects which can be included in the model.

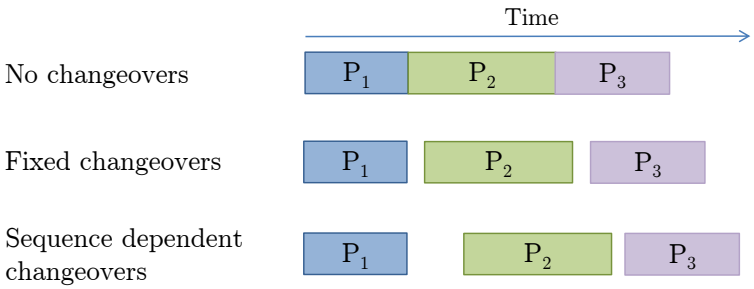


Figure 2.2: Different methods of modelling changeover times between products

The mathematical model must be realistic in order for the results to be meaningful. It must try as closely as possible to mimic what would happen in practice, and thus different versions of the model will be developed as the model evolves in complexity. For the simple case, the whole process (both USP and DSP) can be treated as a black box (Siganporia et al., 2012). However, one of the key limitations of that model is the lack of manufacturing flexibility from coupling upstream and downstream processes to one another. One reason why it is beneficial to decouple upstream and downstream production is because for perfusion processes, manufacturers often completely separate the upstream and downstream process, freezing the intermediate product in between. Allowing the USP and DSP to be modelled separately permits products to be manufactured alongside each other within the same facility, which would not have been possible with a simple black box design.

In theory, the material produced upstream in one facility can be processed downstream in a completely different facility, and thus the model can be adapted for this scenario too.

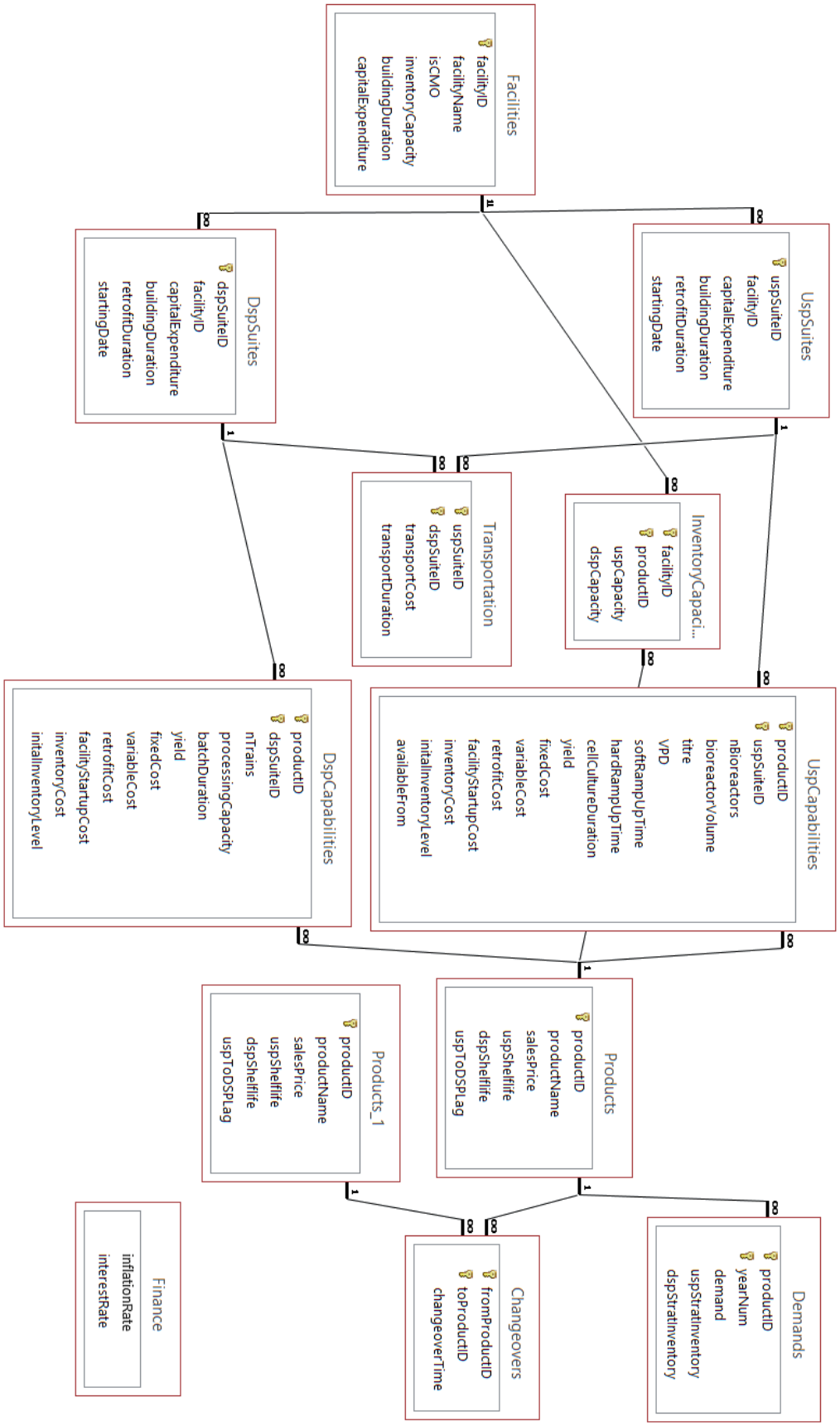


Figure 2.3: Entity relationship diagram depicting input data structure

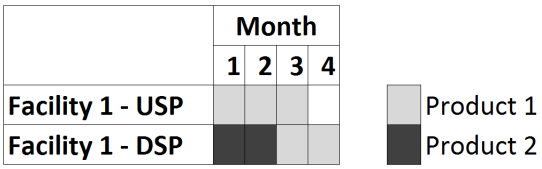


Figure 2.4: Separating USP from DSP. Note that this assumes that there was already some intermediate product stored for product 2.

2.2 Computational Complexity

To create a capacity plan manually, taking all these constraints into consideration, is a very difficult task - but not impossible. The dilemma is that any solution that is found manually is extremely unlikely to be optimal, and in the case of multi-billion dollar biopharmaceutical companies, any sub-optimal solution could be costing them a huge amount in losses. The need for a decision-support tool becomes even more evident when one examines a small test case:

Imagine there are two facilities (i) and two products (p), and that there is a demand for both products at some time in the future. In any given time period, the possible solutions are:

1. p_1 is produced in i_1
2. p_1 is produced in i_2
3. p_2 is produced in i_1
4. p_2 is produced in i_2
5. p_1 is produced in i_1 AND p_1 is produced in i_2
6. p_2 is produced in i_1 AND p_2 is produced in i_2
7. p_1 is produced in i_1 AND p_2 is produced in i_2
8. p_2 is produced in i_1 AND p_1 is produced in i_2
9. No production in either i_1 or i_2

Bearing in mind that this is just for one time period, it becomes easy to see that with a greater number of products and facilities, the problem becomes exponentially more difficult to solve (see Table 2.1).

Table 2.1: Minimum number of theoretical permutations for different cases over 8 years with a time period of one month

No. products	No. facilities	Lower estimate of no. solutions
2	2	10^{91}
4	4	10^{289}
6	6	10^{522}
10	10	10^{1056}

2.3 Framework Structure

This decision-making tool utilises mathematical programming techniques to minimise the manufacturing cost and determine the optimal manufacturing schedule. A mathematical model has been created and a Mixed Integer Linear Programming solver (CPLEX) has been used to optimise the problem.

The mathematical model is written in GAMS code, and as such requires the GAMS base module. GAMS itself cannot solve the problem, but instead relies on external or internal solvers, some of which come free with the base module. For the purposes of this project, however, we will using the CPLEX solver, which can be accessed from within GAMS with the correct licensing, or can be accessed via a GAMS/CPLEX Link (assuming the CPLEX solver has been installed and licensed separately). Should we wish to access the CPLEX optimisation algorithms from an environment other than GAMS (for example, C#), then it is important that we install IBM's ILOG CPLEX Solver separately. The input data that GAMS reads is stored in an Excel spreadsheet, and the variables from the solution are written back into the Excel spreadsheet (this is achieved via the GDXXRW utility, which comes as part of GAMS). Finally, code written in Visual Basic for Applications (VBA) is used within Excel to analyse the data and produce Gantt charts for the schedule. Figure 2.5 shows

the architecture of the framework, and gives examples of the type of data or actions that link components together.

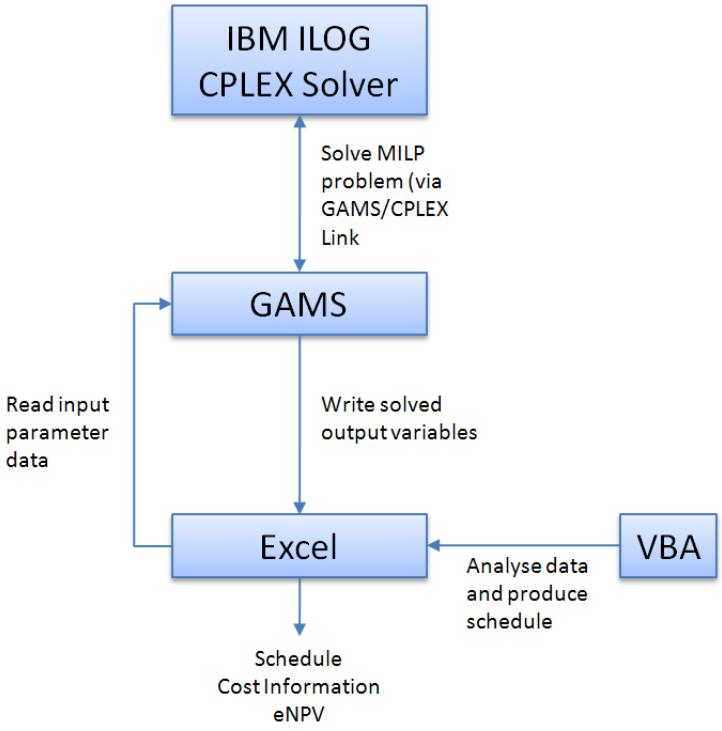


Figure 2.5: System Design

One of the drawbacks of using Excel as the input for GAMS is that editing spreadsheets when changing case studies is a tedious and error-prone process. GAMS must read data as matrices, thus if the number of products or facilities change, the size of the tables in Excel also change, leading to scaling issues. Thus the data was also converted to a relational database format, thereby increasing scalability and ease of use. The entity relationship diagram shown in Figure 2.3 demonstrates how the tables within the database are linked to each other. This change in input format was not completed in time to be incorporated into the framework described here, but it was used in a separate model which used genetic algorithms to optimise production plans. This is explained in more detail later.

2.4 Model Requirements

Having explained the problem in more detail, and how the framework components are structured, it is now necessary to outline what functionality the framework

should be expected to provide. The following are some requirements:

- Gantt chart showing the production schedule
- Number of batches and hence material output per time period
- Facility utilisation
- Customer service level
- COGS
- Capital expenditure
- Net present value

As an example of a Gantt chart, Figure 2.6 shows an 8 year capacity plan for an arbitrary case study with four products and four facilities. In this example, there is no separation of USP and DSP production. The Gantt chart shows the allocation of different products, and allows a user to view the number of batches produced and days used in any given month.

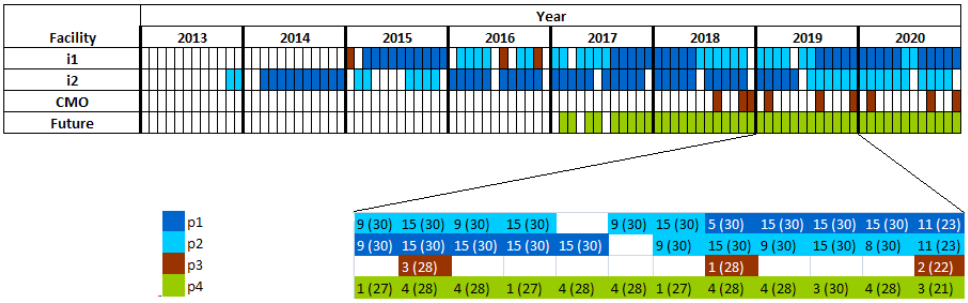


Figure 2.6: Example Gantt chart output including a detailed view of 2019's manufacturing schedule. The numbers represent the number of batches, and in brackets the occupation time. The maximum number of days that can be used in a month is 30.

If the schedule that was obtained via the optimisation is not exactly how a production team would like it, an interface has been developed such that one can tweak the schedule within Excel and execute the GAMS code. In this case, key variables in the mathematical model are fixed, thus the optimisation is very fast,

and acts more like an infeasibility checker. This flexibility can help in instances where there is a bias to manufacturing in a certain way, but where cost is not affected (and hence is not considered in the mathematical model).

2.5 Summary

This chapter has described in more detail the problem being tackled by this decision-support tool. The usefulness of a tool that will aid strategic planning for future products is undoubtedly high, and can enable a company to better predict likely cash outflows owing to capital investment. Uncertainty can be incorporated via running different scenarios, and can help identify more robust manufacturing strategies, and thereby help contain risk.

In the next chapter, a description of the mathematical model is outlined, with results showing how the tool was used on an industrial case study to help determine optimal manufacturing schedules.

Chapter 3

Capacity planning for batch and perfusion bioprocesses across multiple biopharmaceutical facilities

3.1 Introduction

Biopharmaceutical companies with growing portfolios of commercial therapeutics face the challenge of generating medium and long-term production plans for several drugs across several multi-product manufacturing sites that maximise capacity whilst minimising cost.

Production planning is complicated by portfolios of commercial candidates that are made with different cell culture modes: fed-batch mode, or continuous perfusion for labile products. The complication arises from the fact that perfusion cell cultures can span many months, whereas fed-batch cell cultures are usually two weeks in duration. The discrete time representation used in this model is of one month, thus extra modelling constraints need to be introduced to ensure that production is not stopped half-way through a cell culture (since it is meant to model a continuous process). Every time a new perfusion cell culture begins,

ramp-up times need to be considered, since the manufacturer may choose not to harvest any material during this period, due to it not meeting all required specifications. Material is normally harvested semi-continuously from perfusion processes, which is conceptually different from a fed-batch process where material is harvested at the end of the cell culture. Thus these continuous harvests need to be incorporated in the model's constraints. Changing from one mode of operation to another also increases the changeover time normally associated with product switchovers. This adds more complexity to the optimisation since a larger number of constraints is required.

This chapter describes the development of a discrete-time mixed integer linear programming (MILP) model that incorporates both perfusion and fed-batch processes to produce capacity plans and manufacturing schedules. Extra constraints have been incorporated to more realistically model the perfusion process. For example, ramp-up times and cell culture durations spanning multiple time periods have been implemented for perfusion-mode processes. One of the challenges met by this formulation is the ability to include sequence-dependent changeover times between products, which is necessary because switching between perfusion and fed-batch modes takes longer than staying within the same process mode. Annual fixed costs are also included in the model, along with other investment considerations such as retrofitting costs and investment into constructing new facilities. These additional features allow the model to pick strategies based on a more holistic approach, and thus provide more economically feasible solutions. Strategic inventory targets have also been implemented such that the manufacturer can choose to have extra stock of product should demand unexpectedly rise or supply suddenly fall. These extra features add to the complexity of the model, and thus require additional CPU resources in order to obtain a satisfactory solution. Hence a rolling time horizon has been implemented and has successfully managed to improve solutions, whilst at the same time reduce CPU requirements. The impact of variations on key parameters such as demand or titres on the optimal production plans and costs was captured through scenario analysis.

The remainder of this chapter consists of an explanation of the problem domain in Section 3.2, followed by a description of the mathematical formulation used in the MILP model in Section 3.3. An industrial case study is then used to explore the capabilities of the model and identify trends which can be used to aid business decisions. The mathematical nomenclature can be found at the end of this paper.

3.2 Problem Definition

The focus of this work is on long-term multi-site production planning for biopharmaceuticals to minimise the total manufacturing cost and investment whilst satisfying demands. The key features of the problem are discussed below.

3.2.1 Facility features

Allocation of biopharmaceutical facilities across multiple sites requires an understanding of the different facility features such as scale and capability to manufacture each product as well as any differences in fermentation titres and downstream processing yields. The number and size of bioreactors will directly affect a facility's upstream processing (USP) capacity. The same product could be manufactured in two different facilities, with each facility having a different number of bioreactors available, hence the optimisation will select which facility to use based on cost and capacity requirement. Downstream processing (DSP) scales will also vary, since there may be different sized purification equipment such as chromatography columns or filtration rigs. Depending on how the DSP is set up, it could mean that the time required for purification is different between facilities. For example, if the same amount of material is to be processed by a facility with a smaller filtration device, that particular step will be slower (when compared to a larger filtration unit with greater throughput). There may also be multiple DSP trains to process the material from one harvest, which is common for antibody production with high titres (Kelley, 2009). On the other hand, operators could decide to keep the purification time constant, but change the amount of material

processed. These process choices must be correctly captured in the model for there to be realistic manufacturing flexibility.

The capability of a facility to manufacture a product will differ, not just owing to logistical aspects, but also strategic. For example, if a future facility is built with perfusion products in mind, then it may not be possible to later manufacture fed-batch products. A CMO however, may be capable of manufacturing all of the products, but due to licensing and IP issues a company may wish to keep production of certain products in-house. Another key facility feature is the cost of manufacturing a product there. Cost differences are present between in-house and CMO facilities to reflect the extra service cost with CMOs (George and Farid, 2008b).

3.2.2 Fed-batch versus perfusion culture processes

The USP stages of mammalian cell culture processes typically involve either fed-batch or perfusion culture (Pollock et al., 2013). It is also possible to have one or more steps of the seed train as perfusion-mode, and the production cell culture as fed-batch (Pohlscheidt et al., 2013). Perfusion culture is necessary for labile products such as blood factors and enzymes (e.g., Cerezyme) and has also been used for certain stable monoclonal antibody (mAb) products (e.g., Remicade) using retention devices that range from gravity settlers to filtration devices (Pollock et al., 2013). Perfusion processes typically offer higher daily volumetric productivities and hence smaller facility footprints than fed-batch culture strategies. However, they are generally more complex to operate, require increased amounts of media, and are susceptible to higher failure rates (Cacciuttolo, 2007). Newer perfusion retention devices using external tangential flow filters aim to overcome some of these obstacles with the capability to attach to single-use bioreactors combined with lower failure rates and higher productivities (Clincke et al., 2013). This has increased interest in the business case for perfusion-based processes and process economic analyses have explored the cost-benefit of perfusion versus fed-batch processes.23, 25 However, in recent years, fed-batch culture has become the platform choice for most mAbs due to dramatic increases in fed-batch titres

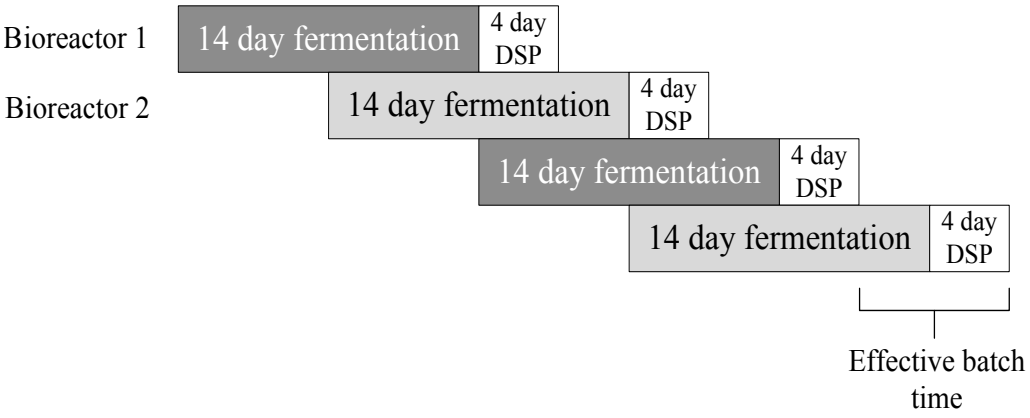


Figure 3.1: Batch-mode process using two staggered bioreactors

combined with ease of operation.

The USP mode of operation has a direct impact on the scheduling of the subsequent DSP steps. In fed-batch mode (Figure 3.1), the culture is harvested at the end of the cell culture duration and subsequently purified by a series of DSP steps (for example, chromatography). In perfusion mode, material is continuously harvested, recovered, captured and sometimes frozen throughout the fermentation culture. Once enough material has been pooled together, it is purified downstream (Figure 3.2). There is also a set amount of time required for quality and assurance tests after each harvest before it can be processed downstream. The ramp-up time is the time required for the cell culture to reach a certain cell density, after which steady state is achieved. Material is sometimes harvested during the ramp-up time, but in this work it is assumed to be discarded. The DSP can be carried out immediately or at a later date, either within the same facility or a different one should there be financial incentive. The DSP can only be carried out immediately after harvesting if no quality release testing is required. Perfusion cell cultures usually operate for longer than fed-batch cultures, and since no clean-in-place (CIP) or steam-in-place (SIP) can occur during this time, there is a greater risk of contamination (Acuna et al., 2011). For perfusion processes, sterility samples are taken every day and viral samples are taken every two weeks. Extra testing may be required for longer cell culture durations since the risk of contamination is increased the longer a bioreactor is operating for.

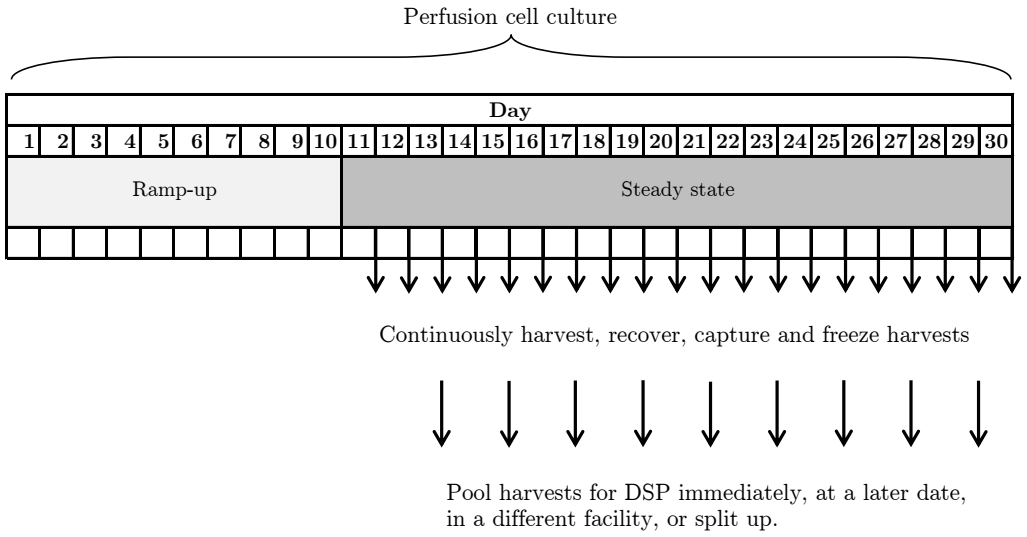


Figure 3.2: Perfusion mode cell culture

Perfusion processes can therefore be modelled as a black box, where DSP directly follows USP, or using a decoupled design. The flexibility in perfusion-mode manufacturing is only apparent if USP and DSP are decoupled from each other. The black box design is simpler and can therefore be solved quicker, but the decoupled design allows for more manufacturing flexibility, which could (depending on input parameters) provide a lower overall manufacturing cost and is also a closer representation of reality.

3.2.3 Key performance indicators

Successful production planning requires consideration of cost factors such as the manufacturing cost, the capital investment required either to build new facilities or retrofit existing ones, as well as inventory costs. The manufacturing cost can be separated into fixed and variable costs. This model assumes the variable cost to consist solely of materials, with costs attributed to labour, depreciation and facility overheads being assigned to fixed costs. The inventory cost includes the actual warehouse costs as well as the cost attributed to the opportunity lost in selling the product. In addition to costs, customer service levels can be assessed to see how much customer demand is met on time. Insufficient capacity will lead to lower customer service levels. It is also important to determine the facility utilisation to avoid idle expenses. Facility utilisation may need to be kept within

certain targets. If facility utilisation is too high, any unplanned downtime could severely affect the customer service level. Under-utilisation, on the other hand, may suggest a misplaced investment in capacity. Together, these performance indicators help a manufacturer to assess the viability of a production plan.

3.3 Mathematical formulation and solution procedure

The following section describes the mathematical formulation developed to address the problem domain. The nomenclature can be found at the end of this chapter. It is important to note that many of the variables have been duplicated for the upstream and downstream parts of the model (for example, the number of batches produced). To aid with legibility, the superscripts 'U' or 'D' denote upstream or downstream respectively. This model uses a discrete time representation, with monthly time resolution. This means, for example, that for an eight year planning horizon there would be 96 time periods. One of the assumptions made in this model is that only one product can be produced in any given time period. This can be justified by the fact that for biopharmaceutical manufacturing, it is unlikely that there is going to be a high rate of switching between products (owing to quality, control, and logistic issues that may arise). So as long as the time periods are relatively small, this limitation of one product per time period should be of no concern. This model uses monthly time periods, hence there should be no issue. Indeed, all the perfusion products in the case study have fermentation cell cultures which last at least than one month, thus are not affected by this limit. One of the reasons this assumption was made was because it made it easier to model the scheduling of the problem, with less computational effort required.

3.3.1 Technical and commercial constraints

Production constraints

In essence, the number of upstream batches produced in time period t, for product p, in fermentation suite i, is denoted by B^U_{ipt} and is equal to the batch rate

(r^U_{ip}) multiplied by the amount of time available (T^U_{ipt}). If there is a changeover between products p' and p, $Z^U_{ip'pt}$ will equal 1, and a campaign changeover time ($\alpha_{p'p}$) is subtracted from the available time. Depending on whether the product is manufactured using fed-batch or perfusion culture, an additional time is subtracted. For perfusion products $p \in P^p$ (Equation 3.1), the ramp-up times (β_p) are subtracted when new perfusion cell cultures begin ($F_{ipt} = 1$). For fed-batch products $p \notin P^p$ (Equation 3.2), the time required for the first batch (τ'_p) is subtracted so that the effective batch rate can be used from that point onwards. For example, if the fed-batch process is like that shown in Figure 3.1, the first upstream batch would take 14 days, but from that point onwards there will be another batch every 7 days. The extra time necessary for the first batch is only required when a new campaign starts ($Y'_{ipt} = 1$). To compensate for the removal of time for the first batch, Y'_{ipt} is added to the number of batches. Hence, when a new campaign of a fed-batch product begins, Y'_{ipt} is equal to 1, and the number of batches produced is equal to 1 plus the effective batch rate multiplied by time available minus time required for the first batch.

$$B^U_{ipt} = r^U_{ip}(T^U_{ipt} - \beta_p F_{ipt} - \sum_{p'} \alpha_{p'p} Z^U_{ip'pt}) \qquad \forall t, p \in P^p, i \in I_t \cap I_p \tag{3.1}$$

$$B^U_{ipt} = Y'_{ipt} + r^U_{ip}(T^U_{ipt} - \tau'_p Y'_{ipt} - \sum_{p'} \alpha_{p'p} Z^U_{ip'pt}) \qquad \forall t, p \notin P^p, i \in I_t \cap I_p \tag{3.2}$$

To ensure only one product is manufactured in a suite at any given time, the binary variables Y^U_{ipt} and Y^D_{ipt}, which are equal to 1 if product p is manufactured in suite i at time t for upstream and downstream suites respectively, are constrained as follows:

$$\sum_p Y^U_{ipt} \leq 1 \qquad \forall\, t,\ i \in I_t \tag{3.3}$$

$$\sum_{p} Y^{D}_{ipt} \leq 1 \qquad \forall\, t,\ i \in I_t \tag{3.4}$$

New upstream campaigns are indicated with Y'_{ipt} being equal to 1, and this can only occur if there was no production of that product in the previous time period.

$$Y'_{ipt} \geq Y^{U}_{ipt} - Y^{U}_{ip,t-1} \qquad \forall t, p, i \in I_t \cap I_p \tag{3.5}$$

The number of upstream batches for products using fed-batch mode is equal to the number of batches in the purification (assuming there is no pooling or splitting of fermentation volumes):

$$B^{U}_{ipt} = B^{D}_{ipt} \qquad \forall\, t,\ p \notin P^{\mathrm{p}},\ i \in I_t \cap I_p \tag{3.6}$$

For perfusion products, the number of downstream batches is simply equal to the batch rate multiplied by the amount of time available:

$$B^{D}_{ipt} = r^{D}_{ip} T^{D}_{ipt} \qquad \forall t, p \in P^{\mathrm{p}}, i \in I_t \cap I_p \tag{3.7}$$

Availability Constraints

In order for production to take place in a facility, it must first be available for use. It may first need to be built, retrofitted, or may even be unavailable for another reason (e.g., being used for another product which is not in the current product portfolio). The variable A^{U}_{ipt} is equal to 1 if facility i is available to product p at time t for upstream production. Variable A^{facility}_{it} is equal to 1 if facility i has been built and is ready to be used at time t, and variable $A^{\text{retrofit,U}}_{ipt}$ is equal to 1 if facility i has been retrofitted for product p and is ready to be used at time t for upstream production.

$$Y^{U}_{ipt} \leq A^{U}_{ipt} \qquad \forall\, t,\ p,\ i \in I_t \cap I_p \tag{3.8}$$

$$Y^{D}_{ipt} \leq A^{D}_{ipt} \qquad \forall\, t,\ p,\ i \in I_t \cap I_p \tag{3.9}$$

$$A^{U}_{ipt} \leq A^{\text{facility}}_{it} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.10}$$

$$A^{D}_{ipt} \leq A^{\text{facility}}_{it} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.11}$$

$$A^{U}_{ipt} \leq A^{\text{retrofit},U}_{ipt} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.12}$$

$$A^{D}_{ipt} \leq A^{\text{retrofit},D}_{ipt} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.13}$$

The availabilities for building a facility or retrofitting ($A^{\text{facility}}_{it}, A^{\text{retrofit,U}}_{ipt}$) are linked to the investment constraints which follow.

Investment Constraints

Before a facility can be used, there must first be investment into the construction of that facility. The facility is not available before the time is takes to construct it (τ^{build}_i). Construction starts as soon as investment is made. The variable K_{it} is equal to 1 if capital is invested at time t.

$$A^{\text{facility}}_{it} \leq A^{\text{facility}}_{i,t-1} + K_{i,t-\tau^{\text{build}}_i} \qquad \forall\, i,\, t \tag{3.14}$$

In order for a product to be manufactured in a facility, any relevant retrofitting must be carried out. L^{U}_{ipt} is equal to 1 if facility i has been retrofitted for product p at time t. The investment for retrofitting must be spent τ^{retrofit}_i time periods before the facility becomes available for that product.

$$A^{\text{retrofit},U}_{ipt} \leq A^{\text{retrofit},U}_{ip,t-1} + L^{U}_{ip,t-\tau^{\text{retrofit}}_i} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.15}$$

$$A^{\text{retrofit},D}_{ipt} \leq A^{\text{retrofit},D}_{ip,t-1} + L^{D}_{ip,t-\tau^{\text{retrofit}}_i} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.16}$$

The model also includes any licence fees and start-up costs, and this is indicated via L_{ipt}. There is no differentiation between upstream and downstream here, since a licence is assumed to be required per facility, not per suite. If there are special licences or costs that are applicable to suites rather than facilities,

then they can be incorporated into the retrofitting costs.

$$A^U_{ipt} \leq A^U_{ip,t-1} + L_{ipt} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.17}$$

$$A^D_{ipt} \leq A^D_{ip,t-1} + L_{ipt} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.18}$$

Fixed Cost Constraints

A simplified fixed cost model is used to calculate the annual fixed cost in each facility. Generally, the products would have different fixed costs, and thus, the annual fixed cost would be the maximum of the fixed costs of the products produced in that year. If no product is manufactured in a given year, then there is still a fixed cost applied because the facility still needs to be maintained under Good Manufacturing Practice (GMP) conditions. Upstream and downstream suite use (U^U_i and U^D_j) is separated so that fixed costs can be attributed individually. If a suite has never been used over the planning horizon (e.g., if it had never been built, or if no product was ever allocated to it), then no fixed costs need to be applied for that suite. Also note that only the facilities which are owned (I^{owned}) need to be subjected to fixed costs. This is achieved in the objective function where the cost is applied.

$$U^U_i \geq Y^U_{ipt} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.19}$$

$$U^D_i \geq Y^D_{ipt} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.20}$$

Timing Constraints

In order to tighten the optimisation's search for an integer number of batches, a minimum processing time can be enforced. The maximum utilisation time in any given month, $T^{\max}_p$, is usually just equal to 30 days, but in some cases this can be adjusted to tighten the optimisation.

$$T^{\min,\mathrm{U}}_p Y^U_{ipt} \leq T^U_{ipt} \leq T^{\max,\mathrm{U}}_p Y^U_{ipt} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.21}$$

$$T^D_{ipt} \leq T^{\max,\mathrm{D}}_p Y^D_{ipt} - \sum_{p'} \alpha_{p'p} Z^D_{ip'pt} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.22}$$

Changeovers occur when there is a product switch within the same facility. In the following equations, $Z^U_{ip'pt}$ is equal to 1 when there is a changeover from product p' to p. If there is an idle period, this model will assume that the changeover will take place in the idle period and thus will not subtract from available production time.

$$Z^U_{ip'pt} \geq Y^U_{ipt} + Y^U_{ip',t-1} - 1 \qquad \forall\, t,\, p',\, p,\, i \in I_p \cap I_{p'} \cap I_t \tag{3.23}$$

$$Z^U_{ip'pt} \leq 1 - Y^U_{ipt} + Y^U_{ip',t-1} - 1 \qquad \forall\, t,\, p',\, p,\, i \in I_p \cap I_{p'} \cap I_t \tag{3.24}$$

$$Z^D_{ip'pt} \geq Y^D_{ipt} + Y^D_{ip',t-1} - 1 \qquad \forall\, t,\, p',\, p,\, i \in I_p \cap I_{p'} \cap I_t \tag{3.25}$$

For perfusion products, new cell cultures start ($F_{ipt} = 1$) when a new campaign starts:

$$F_{ipt} \geq Y^U_{ipt} - Y^U_{ip,t-1} \qquad \forall\, t,\, p \in P^p,\, i \in I_p \cap I_t \tag{3.26}$$

Since perfusion cell cultures have a fixed length (τ^T_p), it is necessary to ensure that a new cell culture is started once the previous one has finished.

$$F_{ipt} \geq Y^U_{ipt} + F_{ip,t-\tau^T_p} - 1 \qquad \forall\, t,\, p \in P^p,\, i \in I_p \cap I_t \tag{3.27}$$

The following constraint ensures that the perfusion campaign is run for its entire length, and that each day in the month is also used. This last point is important, since once a perfusion process has started, it should be run continuously, and thus there cannot be idle days in the middle of the cell culture. The cell culture's duration in days and time periods are represented by τ_p and τ^T_p respectively. Thus, if a new 150 day cell culture is started, F_{ipt} will be equal to 1, and so the equation forces the total time used during the cell culture to be equal to 150 days. Note that although the equation does not explicitly restrict

the total time, it is limited in the timing constraint from earlier (Equation 3.21).

$$\tau_p F_{ipt} \leq \sum_{\theta=0}^{\tau_p^T - 1} T^U_{ip,t+\theta} \qquad \forall\, t,\, p \in P^p,\, i \in I_p \cap I_t \tag{3.28}$$

The following constraint is needed to prevent the situation where perfusion campaigns are started near the end of the planning horizon, without enough time to finish. It also solves a problem where F_{ipt} can potentially be equal to 1 even if it is not the beginning of a new perfusion campaign (this can happen if the model wishes to add downtime to lower the cost or meet a constraint).

$$F_{ipt} = 0 \qquad \forall\, i,\, p \in P^p,\, t \in T : t > (|T| - \tau_p^T + 1) \tag{3.29}$$

Inventory Constraints

The constraint shown in Equation 3.30 states that the inventory level for the fermentation product (I^U_{ipt}) is equal to its previous level plus any material produced in subsequent batches (taking into consideration quality checks of duration τ_p^{qc}), minus any material which is used for purification (Q_{ijpt}). The amount of material produced in one time period is equal to the output per batch (x^U_{ip}) multiplied by the number of batches, and is adjusted using a rejection coefficient (R). So if 5% of material is rejected, then 95% of the material from the batches can enter the inventory.

$$I^U_{ipt} = x^U_{ip}(1-R)B^U_{ip,t-\tau_p^{\text{qc}}} + I^U_{ip,t-1} - \sum_j Q_{ijpt} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.30}$$

The flow of material from the fermentation suite, i, to the purification suite, j, is characterised by Q_{ijpt}. As previously mentioned, the lot size for the purification train is fixed for each product, and this is enforced by the following constraint:

$$\sum_i Q_{ijpt} = x_p^{\text{load}} B^D_{jpt} \qquad \forall\, t,\, p,\, j \in I_t \cap I_p \tag{3.31}$$

where i is the fermentation suite and j is the purification suite. Equation 3.31

states that the total flow of material in a given time period from all the fermentation suites to the current purification suite must equal an integer number of batches multiplied by the batch lot size. This constraint means that material can be pooled from different fermentation suites and processed as one batch in a DSP suite. This is an assumption in the model, and should be adapted if pooling is not allowed.

The downstream inventory level of product p in time period t in facility i is equal to the amount produced (taking into consideration production losses) plus the previous month's inventory level, minus any amount of material sold (S_{ipt}) or wasted (W_{ipt}). The amount sold is limited by demand (Equation 3.39). Assuming all material here is used, the amount produced is simply equal to the output per batch (x^D_{ip}) multiplied by the number of batches (B^D_{ipt}).

$$I^D_{ipt} = x^D_{ip} B^D_{ipt} + I^D_{ip,t-1} - S_{ipt} - W_{ipt} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.32}$$

In any given time period, the model will try to maintain the strategic inventory level ($I^{\min}_p$) by calculating the gap between the inventory level and the target ($I^{\text{under,U}}_{pt}$), and then penalising this variable in the objective function.

$$\sum_i I^U_{ipt} \geq I^{\min,U}_p - I^{\text{under,U}}_{pt} \qquad \forall\, p,\, t \tag{3.33}$$

$$\sum_i I^D_{ipt} \geq I^{\min,D}_p - I^{\text{under,D}}_{pt} \qquad \forall\, p,\, t \tag{3.34}$$

Utilisation Constraints

There are maximum utilisation targets for in-house facilities, and thus constraints need to be put into place to accomplish this. For every in-house facility and each year, the following equations restrict the total time used for each product in each month of the year to be below the maximum allowed. Therefore, if the maximum desired facility utilisation is 75%, $T^{\text{max util}}$ can be set to 270 days. The model

applies the same utilisation target to both upstream and downstream suites.

$$\sum_{p} \sum_{t \in T_y} T^U_{ipt} \leq T^{\text{max util}} \quad \forall i \in I^{\text{owned}}, y \tag{3.35}$$

$$\sum_{p} \sum_{t \in T_y} T^D_{ipt} \leq T^{\text{max util}} \quad \forall i \in I^{\text{owned}}, y \tag{3.36}$$

Shelf-Life Constraints

The products have a limited shelf-life (ζ_p), and so a constraint needs to be introduced (Equation 3.38) to ensure that the product is sold before its shelf-life expires. Also, the intermediate product from the upstream process must be purified before it expires (Equation 3.37).

$$I^U_{ipt} \leq \sum_{j \in P_j} \sum_{\theta=1}^{\zeta^U_p} Q_{ijp,t+\theta} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.37}$$

$$I^D_{ipt} \leq \sum_{\theta=1}^{\zeta^D_p} S_{ip,t+\theta} \qquad \forall\, t,\, p,\, i \in I_t \cap I_p \tag{3.38}$$

Sales Constraints

In order to allow for feasible solutions in situations where demand cannot be met, a backlog variable Δ_{pt} is introduced. This variable is then penalised in the objective function so as to ensure as much demand is met as possible. Some products (notably those which use perfusion) require quality checks before being passed to purification, and thus this time must be considered when meeting the demand. If the demand for a certain product is in month 8, but it takes 1 month to perform the quality checks, then the material must be ready by month 7, ensured by $S_{ip,t-\tau_p^{\text{quality}}}$.

$$\sum_{i \in I_p} S_{ip,t-\tau_p^{\text{quality}}} = D_{pt} - \Delta_{pt} + \Delta_{p,t-1} \qquad \forall\, p,\, t \tag{3.39}$$

3.3.2 Objective function

The discount factor is calculated as:

$$\epsilon_t = \left(\frac{1+f}{1+g}\right)^{t-1} \tag{3.40}$$

where f and g are the inflation and interest rate respectively.

The individual costs have been broken down as follows:

$$\text{Inventory cost} = IC = \sum_i \sum_p \sum_t \epsilon_t \big(\rho_{ip}(I^U_{ipt} + I^D_{ipt}) + \rho^{\text{carry}}_{ip}(I^U_{ipt} + I^D_{ipt})\big) \tag{3.41}$$

$$\text{Inventory penalty cost} = IPC = \sum_p \sum_t \epsilon_t \big(I^{\text{penalty}}_{ip}(I^{\text{under},U}_{ipt} + I^{\text{under},D}_{ipt})\big) \tag{3.42}$$

$$\text{Variable cost} = VC = \sum_i \sum_p \sum_t \epsilon_t \big(\eta^{\text{bias}}_{ip}(\eta^U_{ip} B^U_{ipt} + \eta^D_{ip} B^D_{ipt})\big) \tag{3.43}$$

$$\text{Fixed cost} = FC = \sum_{i \in I^{\text{owned}}} \sum_t \epsilon_t \big(u^{\text{cost},U}_i U^U_i + u^{\text{cost},D}_i U^D_i\big) \tag{3.44}$$

$$\text{Transportation cost} = TC = \sum_i \sum_j \sum_p \sum_t \epsilon_t (q^c_{ij} Q_{ijpt}) \tag{3.45}$$

$$\text{Waste cost} = WC = \sum_i \sum_p \sum_t \epsilon_t \big(w^{\text{cost}} W_{ipt}\big) \tag{3.46}$$

$$\text{Backlog penalty cost} = BPC = \sum_p \sum_t \epsilon_t \big(\delta_p \Delta_{pt})\big) \tag{3.47}$$

$$\text{Facility investment} = FI = \sum_i \sum_t \epsilon_t \big(\kappa_i K_{it}\big) \tag{3.48}$$

$$\text{Retrofitting cost} = RC = \sum_i \sum_p \sum_t \epsilon_t \big(\lambda^U_{ip} L^U_{ipt} + \lambda^D_{ip} L^D_{ipt}\big) \tag{3.49}$$

$$\text{Licence cost} = LC = \sum_i \sum_p \sum_t \epsilon_t \big(\lambda_{ip} L^U_{ipt}\big) \tag{3.50}$$

The total cost consists of all the above costs summed together, and finally equations 3.1-3.51 form the MILP problem to be optimised.

$$\text{Minimise Total Cost} = IC + IPC + VC + FC + TC + WC + BPC + FI + RC + LC \tag{3.51}$$

3.3.3 Optimisation Strategies

To obtain a good solution within reasonable time becomes increasingly more difficult as the number of products, facilities or time periods being captured rises. In order to achieve better solutions, a rolling time horizon was used, whereby a smaller optimisation problem was run first, and part of the solution to this sub-problem was used to initiate the subsequent larger problem. For example, if the capacity plan was for 8 years, the first sub-problem could be a 4 year plan, and once this has solved the second sub-problem could be 5 years in length, but with the binary variables in the first year fixed to the solution from the previous sub-problem. The next sub-problem would be 6 years in length, with the first 2 years fixed from previous solutions, and the process continues until the full 8 years has been captured. Although this approach is unlikely to find the true optimum (since optimality gaps are accumulated for each sub-problem), for the example investigated in this work it can provide better solutions than the full scale optimisation under finite time. It should be stressed that given an unlimited amount of time, the full scale mode will always provide the best solution. Figure 3.3 shows a rolling time horizon where only four years are actually optimised in any given sub-problem, with the time horizon expanding by one year each time, fixing the binary variables of earlier years using the solution from the previous sub-problem. The rolling horizon approach implemented in this work avoids infeasible situations by allowing backlogs to accumulate if demands in future years are greater than the model was previously able to detect in the sub-problems. Backlogs are penalised in the objective function, hence inferior solutions could arise. However, in the base case presented here, the rolling time horizon approach performed better than the full model in finite time.

	Year							
Sub problem	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
1								
2								
3								
4								
5								

Figure 3.3: Illustration of a rolling time horizon. Dark grey boxes show years where optimisation takes place, white boxes where no optimisation occurs, and light grey boxes where the binary variables are fixed from the previous solution.

3.4 Illustrative Example

3.4.1 Input Data

This framework is tested on a case study of a generic portfolio of four drugs and four facilities, key details of which are listed in Tables 3.1-3.4. Representative data for this case study were derived from literature sources (for example, (Marichal-Gallardo and Álvarez, 2012)) as well as through discussions with industrial practitioners involved in fed-batch and perfusion processes as well as production planning. The four facilities consist of two in-house facilities, one contract manufacturer, and one facility that can be built in the future if required. The four products are in differing stages of clinical trials, but the demands modelled here are for when the products reach the consumer market. Hence the points where demands start in Table 3.3 differ according to how close the product is to market penetration. The time horizon for this case study is eight years.

The quality control / quality assurance (QC/QA) time shown in Table 3.1 is only applicable to perfusion processes, whereby the intermediate frozen material from fermentation is checked prior to purification. This can lead to a substantial lag between material being produced in the fermentation step and it being able to be purified and thus meet demand, hence is included in the model.

Not all products can be manufactured in every facility, and for those combinations which are allowed there may be a one-off retrofitting cost associated with initial production. For example, for strategic reasons a company may wish to keep the production of one of their products to in-house facilities only, and thus

Table 3.1: Process data for drugs

	Product			
	p_1	p_2	p_3	p_4
Process data				
USP				
Fermentation mode	Perfusion	Perfusion	Perfusion	Fed-batch
Cell culture duration (days)	150	60	28	14
Ramp-up time (days)	10	10	10	-
Harvest (AU[a]/day)	14.3	37.8	4.8	-
Shelf-life (months)	24	24	24	-
QC/QA time (days)	30	30	4	-
DSP				
Lot size (AU[a])	450	1000	720	105,000
Duration (days)	1.5	1.5	4	4
Shelf-life (months)	24	24	24	24
Cost data				
USP				
Variable (RMU[b]/AU[a])	0.05	0.05	0.225	0.018
Fixed (RMU[b]/year)	65	65	65	65
DSP				
Variable (RMU[b]/AU[a])	0.002	0.002	9000	100
Fixed (RMU[b]/year)	48	48	48	48
Sales price (RMU[b]/AU[a])	6	6	27	0.1

[a] Arbitrary units [b] Relative monetary units

CMOs would not be available for its manufacturing. In order to use in-house facilities however, retrofitting is required, which must be taken into account during the optimisation. Other products may not be able to be manufactured in a facility simply because the correct equipment is unavailable and retrofitting may be infeasible. Table 3.2 shows the production relationships between the products and facilities in this case study, and also states which combinations require retrofitting.

Table 3.3 shows what the desired inventory levels for the intermediate frozen material and final DSP products are, and it is assumed that these levels remain constant throughout the 8 years of capacity planning. In reality, these figures would probably change, since they are influenced by annual demand, and thus as

Table 3.2: Facility manufacturing capabilities. Note: Product can (Y) or cannot (N) be produced in facility. Retrofitting requirement denoted by *. Facilities i_1, i_2 and the future facility are owned.

	Product			
Facility	p_1	p_2	p_3	p_4
i_1	Y*	Y*	Y*	Y*
i_2	Y	Y	N	N
CMO	N	N	Y	Y
Future	Y	Y	N	Y

Table 3.3: Product demand and strategic inventory levels (arbitrary units, $\times 10^3$)

	Year								Strat. Inv.	
Product	1	2	3	4	5	6	7	8	USP	DSP
p_1	0	20.2	20.3	20.5	21.4	27.2	28.3	29.9	8.6	26.4
p_2	0	0	1.1	3.2	5.3	7.4	9.5	11.5	22.7	19.2
p_3	0	0	0	0.4	0.4	0.4	0.44	0.48	2.2	0.2
p_4	0	0	0	0	2500	2750	3030	3330	-	1900

demand increases over the years so would the strategic inventory level.

Once a product has shown promise and the company wishes to expand to commercial manufacturing, a biologic licence application (BLA) and prescription drug user fee (PDUF) needs to be applied for, which can total just over $2M (Kux, 2012). Each time a product is manufactured in a new facility, a licence application needs to be submitted, and thus the model will try to minimise the number of licences applied for and keep production limited to one facility if possible. Table 3.4 shows the different costs associated with starting production in a particular facility for a certain product. These costs include the licence costs mentioned previously and retrofitting costs (new equipment and facility utilities).

There are also changeover times between the products, as listed in Table 3.5, which are important to model, since when the process mode changes from perfusion to fed-batch, there can be large amounts of downtime due to swapping large unit operations which cannot be shared.

Table 3.4: Initial start-up costs (including retrofitting, CMO negotiation fees, and licences) in relative monetary units.

		Product			
		p_1	p_2	p_3	p_4
USP	i_1	32.5	32.5	32.5	32.5
	i_2	0	0	-	-
	CMO	-	-	7	7
	Future	10	10	-	10
DSP	i_1	87.5	87.5	87.5	87.5
	i_2	0	0	-	-
	CMO	-	-	7	7
	Future	10	10	-	10

Table 3.5: Changeover times between products. The units are in days, and represent the time taken to change from product p' to p.

		Product (p)			
		p_1	p_2	p_3	p_4
Product (p')	p_1	7	7	7	14
	p_2	7	7	7	14
	p_3	7	7	7	14
	p_4	14	14	14	7

3.4.2 Computational Results

The model optimised the production plan of the four products across four available facilities. The results in Figure 3.4 show the plan over 8 years for different demand scenarios. The base case requires the use of a CMO to meet the demand for p_3 and excess demand for p_4. Note that the manufacturing of a product is kept within one facility if possible so as to minimise licence fees. It is clear to see that when the demand is low, all production can be met in-house and without further expansion. Higher demands require almost full use of all the facilities available, and expansion to a CMO and new facility. Despite not being shown here, the market demands for all the products were met in full for almost all scenarios (100% customer service level). Only in the last year of the +50% demand case was there a small backlog for p_3 and p_4 (customer service level of 95%). Therefore, from a strategic viewpoint, the scenario with higher demand looks less robust since the facilities are heavily utilised and there are already small backlogs accumulating. There is very little margin for error should there be a contamination or failed batches, thus extra capacity would be desirable.

A cost breakdown for the three demand cases shown in Figure 3.5 was conducted and shows a clear increase in cost attributed to CMO activity in the higher demand case (Figure 3.5c). For in-house production, the ratio between variable to fixed costs ranges from approximately 1:7 (low demand) to 1:4 (high demand). This range is justifiable, since as the demand increases, so too will the variable costs, whilst the fixed costs will remain unchanged. It should be noted that for this particular case study, once production in a facility has started, annual fixed costs will be applied to that facility from that point onwards, because most activities included in the fixed costs (such as labour, facility maintenance and cleaning) will be on-going even if there is an idle year. The higher demand case also shows that 5% of the total cost comes from the investment required to build the new facility.

Capital expenditure information for all three demand cases is shown in Figure 3.6, and correlates to the retrofitting and facility investment costs in Figure 3.5.

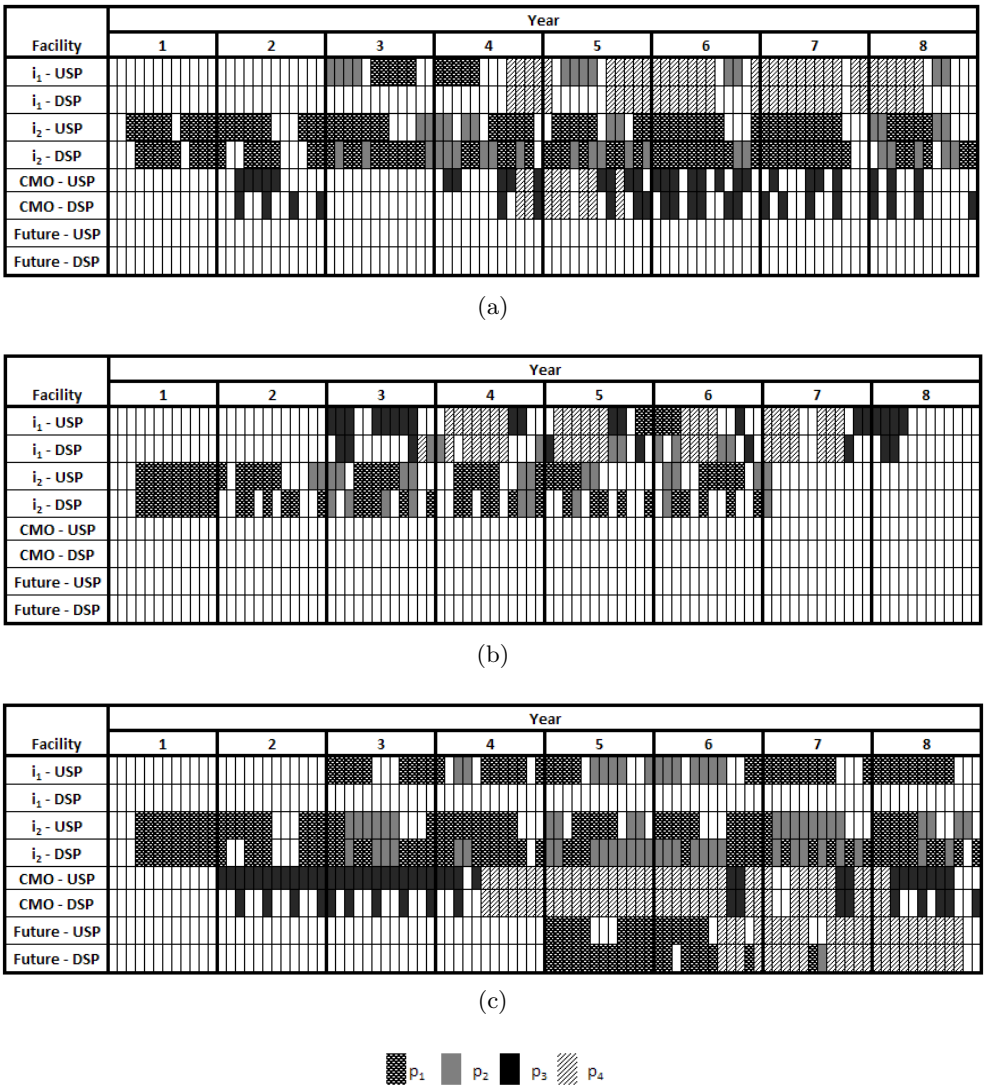


Figure 3.4: Manufacturing schedule for (a) the base case , (b) -50% demand and (c) +50% demand.

The capital expenditures for the reduced demand case and base case are only that of retrofitting. For the higher demand scenario a new facility is required, and the cost of building the facility is spread out over four years, hence the expenditure between years one to four. Retrofitting costs are minimised by the model attempting to keep production within one facility if possible.

Another scenario that may occur is variability in titres for certain products. Process parameters for products in early stages of development are not as well-known as process parameters for commercial products or products in late stage development. Also, when approaching a CMO they may have superior technolo-

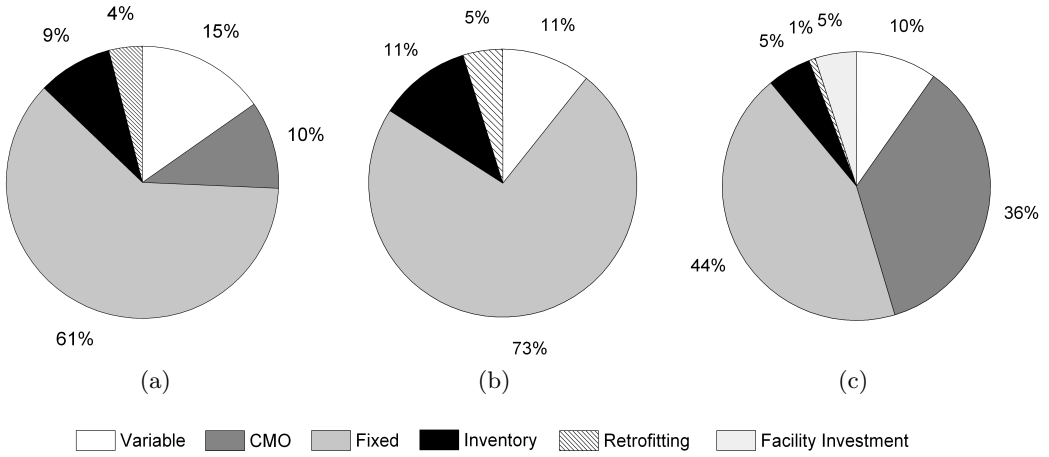


Figure 3.5: Cost breakdown for different demand cases. Base case is shown in (a), 50% decrease in demand in (b) and 50% increase in demand in (c).

gies which can boost titres. Scenarios were carried out to see how varying the titres for p_3 and p_4 by a 25% reduction and 50% increase could affect the capacity requirement (Figure 3.7). These two products were chosen because less was known about their manufacturing processes since they were in early clinical trials. Products p_1 and p_2, on the other hand, were nearing the end of their trials, hence process parameters are known with greater certainty. The reason the titre is varied from -25% to +50% is based on the assumption that if there were to be titre changes/fluctuations, it is more likely to be in the positive direction due to ongoing research, improving cell lines or process design. There is still the risk, however, that the process may not scale well, and hence lower titres are also examined. Titre variations of $\pm 20\%$ are not uncommon when scaling up a process (Amanullah et al., 2010). For example, a lower titre cell line may be selected if it generates fewer host cell impurities or demonstrates more consistent behaviour. When the titre is lower than expected, a much larger proportion of external capacity is required, both in the form of a CMO and through building a new facility. The choice of whether to go to a CMO or build a future facility is mainly influenced by cost, and is discussed later on. Note though, that under the base case conditions a CMO is the preferred choice, since there is much less capital investment required, and the fixed overheads (which are the dominant costs for in-house production) are no longer applied in the same way as for in-

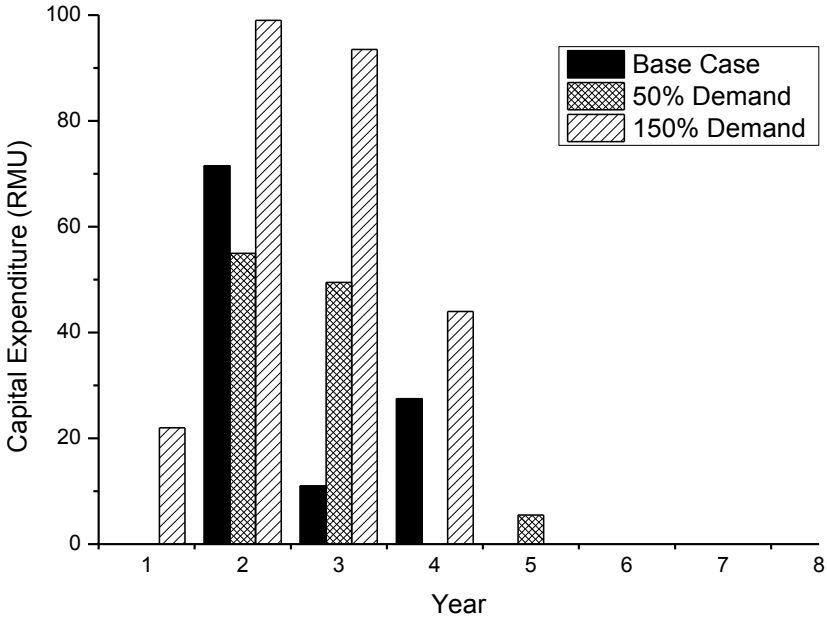


Figure 3.6: Capital expenditure profiles for the base case, 50% demand, and 150% demand.

house facilities. The CMO would still pass on its fixed costs to its customers, but if manufacturing does not span an entire year and the CMO has other clients, this will amount to less than would have otherwise been spent in-house. A CMO alone would not have been enough to meet demand for the reduced titre scenario, hence the future facility was required. With higher titres, the model pushes for a greater proportion of in-house manufacturing (75%), since there is now unused capacity in the existing facilities.

This case study includes a CMO in the list of available facilities, and as such the costs of production there will be different to the in-house production costs shown in Table 3.1. The cost of production in a CMO can be up to three times greater than in-house manufacturing, depending on the scale of production (Kelley, 2009). In the base case we have stated that the CMO costs are 50% higher. However, this is only an assumption, and hence the model was used to show what would happen to the capacity plan if the CMO costs were to change. CMOs are naturally more expensive than in-house production, since they not only need to cover their costs but also charge commission. The extra amount that is paid will be dependent on the CMO's experience, location and technology it has to offer. Given that the CMO costs in this particular case study are

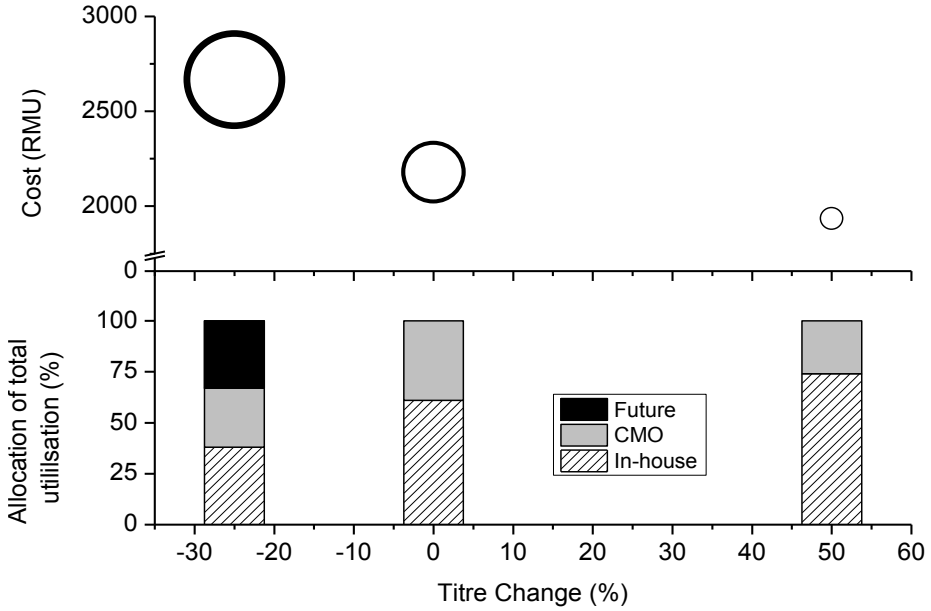


Figure 3.7: Cost and utilisation vs. titre variance. The size of the bubbles represent the amount of extra capacity required to meet the demand. This extra capacity could be sourced from a CMO, or a future facility. The utilisation percentage used for this figure relates to the USP only.

uncertain, an analysis was conducted to see how much more expensive the CMO had to become before it became cheaper to build a new facility and produce in-house. Figure 3.8 shows that once the CMO becomes 50% more expensive than in-house production, a future facility provides alternative means of meeting market demand at lower costs. The utilisation of the CMO decreases as the cost of the CMO increases, but it never reaches 0% (even at 10 times the cost of in-house production) because there is simply not enough capacity in the existing facilities for the fermentation of p_3 (which is being produced in the CMO). On top of this, the future facility cannot produce p_3, hence the fall in utilisation for the CMO is not as large as one would initially expect.

Inventory profiles are useful to see whether the results are what one would expect, since they clearly show whether the targets are being met and if there is a lot of variation. Figure 3.9 shows the inventory profile for a perfusion-mode process, and thus includes the upstream inventory level as well as the downstream level. The figure also includes the strategic inventory levels that should be maintained throughout the capacity plan. As noted before, the levels may in reality change over time, but this model assumes them to remain constant.

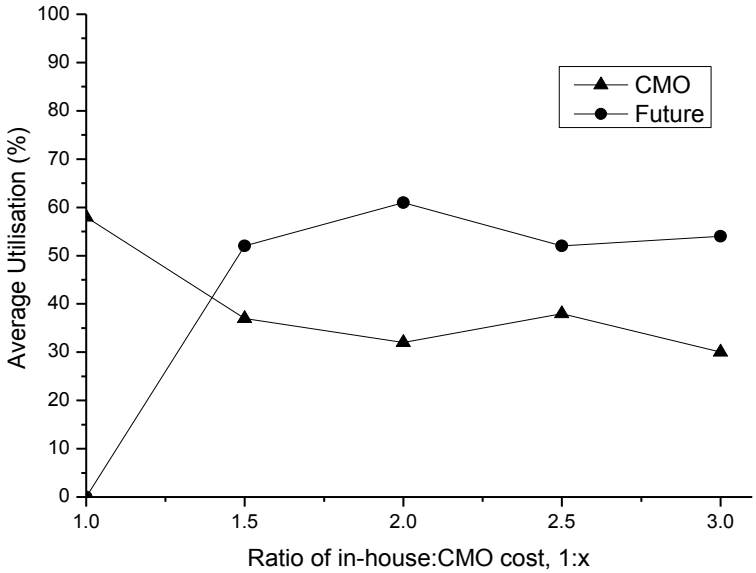


Figure 3.8: Utilisation of CMO facilities will decrease as its cost goes up, meaning more production goes to in-house manufacturing. Since the existing in-house facilities are already near full capacity in the base case, a future facility is built in order to cope with demand. The utilisation percentages displayed here are for the USP production and are the average monthly utilisation from the moment the facility is used to the end of the eight year capacity plan.

The figure quickly demonstrates to a manager that the correct inventory levels are being maintained in the middle of the plan, but near the end the levels tend to drift downwards towards zero. This is actually owing to the fact that the penalty applied for being under the strategic level in the objective function is applied on a monthly basis, and thus near the end there are fewer months available to penalise the shortfall, hence being under the strategic level no longer has such a detrimental effect on the objective function. It therefore becomes cheaper to have less product in storage.

Utilisation graphs can also be used to detect if extra capacity could be directed towards an existing facility with low utilisation, or whether a facility is deemed to be utilised too much and hence raises risk concerns should there be any unplanned downtime. Figure 3.10 shows how facility i_2 is almost at maximum capacity, with only small breaks in production. The breaks in production are actually there on purpose, since there is a utilisation cap of 75%. This provides leeway should problems with failed batches occur. Facility i_1 still has some available capacity, but not enough to meet all demand, hence why the CMO is used in the base case.

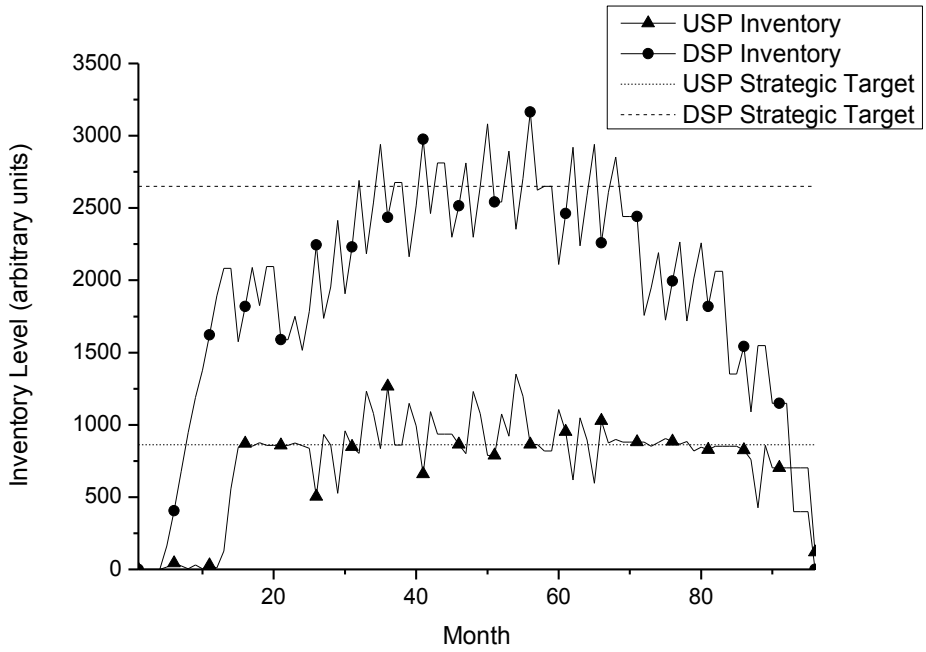


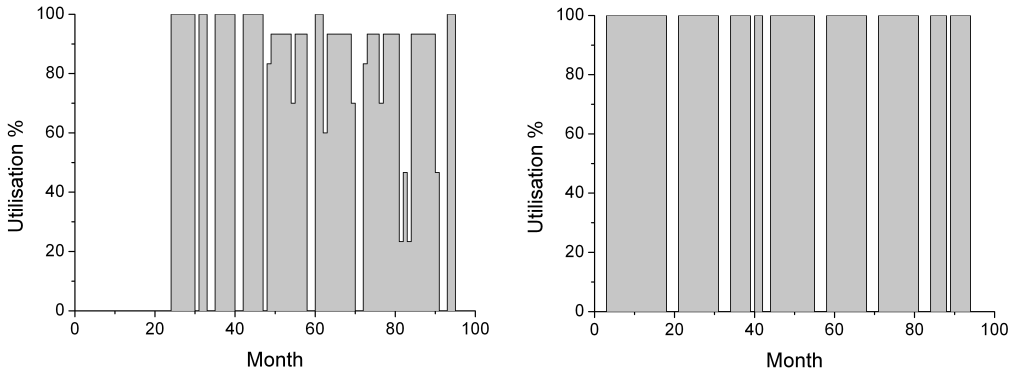
Figure 3.9: Inventory profile for p_1, including both USP and DSP levels. The optimisation attempts to maintain strategic levels, but it will always place more importance on meeting demand first.

3.4.3 Computational Statistics

The optimisation was performed on an Intel Xeon W3565 Quad-core 3.2GHz processor, with 6GB RAM running Microsoft Windows XP 64-bit. The framework presented in this paper uses the CPLEX 12.5.1 solver (Corporation, 2012) within GAMS 24.1.3 (Rosenthal, 2011) to solve the MILP problem, and outputs the solution to Microsoft Excel for analysis using Visual Basic for Applications (VBA). All optimisations (full-scale and rolling time horizon sub-problems) were completed to within 5% optimality. The optimality gap is defined as:

$$\text{Optimality gap} = \frac{\text{Best theoretical objective} - \text{Best feasibile objective}}{\text{Best feasible objective}} \tag{3.52}$$

Here, the best theoretical objective is the solution obtained when the model is relaxed and no longer has to abide by integer constraints. This is of course not necessarily going to provide a feasible solution, but is a limit to how good the solution could be if there were no integer constraints. The best feasible solution



(a) (b)

Figure 3.10: Monthly utilisation charts for in-house facilities (a) i_1 (a) and (b) i_2 for the base case. The percentages are of all products aggregated together for the fermentation (USP). Note that i_1 cannot be used for the first two years since during that time another product (not modelled here) has been designated to it.

is one where integer constraints are in place. A 5% optimality gap means that the objective value of the resulting solution is 5% away from the theoretical best if there were no integer constraints. It may be the case that the solution is actually the best solution available, but cannot be proven by the solver.

Although the case study outlined previously is relatively small, in that it only consists of four products and four facilities (each with upstream and downstream suites), the problem itself is computationally difficult to solve. Table 3.6 shows how the number of variables and constraints in the model increases substantially with increasing numbers of products, facilities and time periods. It should be noted that these numbers would fluctuate depending on the individual case. For example, if a product cannot be manufactured in a certain facility, then a set of

Table 3.6: Model statistics for various numbers of products (p), facilities (i) and time periods (t). The case study presented in this paper is highlighted in bold.

Case	Constraints	Continuous Variables	Discrete Variables
$2p, 2i, 48t$	6,605	4,719	1,148
4p,4i,96t	**33,183**	**25,037**	**5,184**
$8p, 8i, 192t$	426,567	350,349	45,696

Table 3.7: Comparison between the computational results for the full scale problem and the rolling time horizon. Note: In the full scale model all eight years were planned for simultaneously. In the rolling time horizon approach either 3 or 4 years were being optimised whilst expanding the horizon by one year for each sub-problem. The time reported for the rolling horizon approach is the sum of all sub-problems. The optimality gaps shown for the rolling time horizons are calculated based on the best bound from the full scale model. Each sub-problem was optimised to within 5% optimality.

Case	Obj. Func. (min)	Optimality gap (%)	CPU sec
Full Scale	3389	13.1	10,800
Rolling 3/1	3467	15.1	140
Rolling 4/1	3327	11.5	1,484

constraints and variables would be eliminated. The statistics for the case study presented in this paper are represented by the bold highlighted case in Table 3.6. It is clear that the model could become considerably larger in size if just a few extra products or facilities were added to the case study. Hence, as the problem size increases it becomes more critical to adopt solution strategies that make the problem tractable, such as a rolling time horizon.

Table 3.7 shows a clear improvement in using a rolling time horizon, both in terms of obtaining a better optimal solution and also a reduction in CPU time. Obviously, by solving multiple sub-problems (each to a 5% optimality gap), the best bound in the final sub-problem will have accumulated a divergence from the full scale problem, hence for comparison the best bound for the full scale problem is used for calculating all the optimality gaps. The 3/1 rolling horizon approach seems to offer the most in terms of computational speed, whereas the 4/1 approach finds a better solution but at the cost of extra computational effort. Compared to an optimisation of 3 hours with the full scale model, the 4/1 rolling horizon provides a better solution within much less time.

3.5 Summary

This chapter has demonstrated how production plans for fed-batch and perfusion bioprocesses can be optimised using mathematical modelling by incorporat-

ing various costs and time constraints, including sequence-dependent changeover times. Both the upstream and downstream processes have been incorporated into the model, and decoupled by the use of an intermediate storage step, which allows greater flexibility for perfusion processes. The results demonstrate how capacity plans can be quickly determined for various scenarios, aiding the manufacturer in deciding when to consider outsourcing production, and the capital expenditure likely to be required.

The solutions acquired using this framework were improved through a rolling time horizon solution procedure, and the CPU time required was also substantially reduced. Future work will include incorporating features to maintain strategic inventory levels throughout the time horizon, addressing multiple objectives, and reducing the optimality gap even further by appropriate model reformulations.

3.6 Nomenclature

Indices

$i,\ j$	facility (alias)
$p,\ p'$	product (alias)
$t,\ \theta$	time period (alias))
y	year

Sets

I	faclities
I_p	facilities which produce product p
I_t	facilities available in time period t
I^{owned}	owned facilities
P	products
P^{p}	perfusion products
P_i	products which are produced by facility i
T	time periods
T_y	time periods in year y
Y	years

Scalars

H	time horizon (days)
R	rejection coefficient
τ^{retrofit}	retrofitting time)
w^{cost}	waste cost

Parameters

$\alpha_{p'p}$	changeover time from product p' to product p (days)
β_p	ramp-up time (perfusion only)
δ_p	backlog penalty cost
ϵ_t	discount factor
ζ_p^U	upstream product shelf-life
ζ_p^D	downstream product shelf-life
η_{ip}^U	upstream product batch cost
η_{ip}^D	downstream product batch cost
κ_i	facility investment cost
λ_{ip}	licence fees
λ_{ip}^U	retrofitting cost for upstream product
λ_{ip}^D	retrofitting cost for downstream product
ρ_{ip}	storage cost
ρ_{ip}^{carry}	carry of inventory cost
τ_p'	duration of first batch of a fed-batch process (days)
τ_p	perfusion cell culture duration (days)
τ_p^T	perfusion cell culture duration (time periods)
τ_i^{retrofit}	time taken to retrofit facility (time periods)
τ_i^{build}	time taken to build facility(time periods)
τ_p^{qc}	time required for QCQA (time periods)
$T_p^{\max,U}$	maximum USP production time available within time period
$T_p^{\max,D}$	maximum DSP production time available within time period
D_{pt}	demand of product p at time period t (units vary)
I_p^{penalty}	penalty applied when strategic inventory is not met
$I_{pt}^{\text{min,U}}$	USP strategic inventory level
$I_{pt}^{\text{min,D}}$	DSP strategic inventory level
$n_{ip\theta}^{\rho}$	number of USP batches that are produced in period θ of cell culture
q_{ij}^{cost}	cost to transport intermediate material from facility i to j
r_{ip}^U	USP batch rate (batches/day)
r_{jp}^D	DSP batch rate (batches/day)
$u_i^{\text{cost,U}}$	USP fixed cost
$u_j^{\text{cost,D}}$	DSP fixed cost
x_p^{load}	downstream lot sizes (units vary)
x_{ip}^U	USP batch output (units vary)
x_{jp}^D	DSP batch output (units vary)

Binary Variables

F_{ipt}	1 if there is a new perfusion culture
Y_{ipt}^U	1 if product p is produced in suite i over period t (USP)
Y_{ipt}^D	1 if product p is produced in suite i over period t (DSP)

Integer Variables

B_{ipt}^U	number of USP batches produced in suite i over time t of product p
B_{ipt}^D	number of DSP batches produced in suite i over time t of product p

Positive Variables

A^U_{ipt}, A^D_{ipt}	1 if USP or DSP suite is available
$A^{\text{retrofit,U}}_{ipt}$	1 if USP suite has been retrofitted
$A^{\text{retrofit,D}}_{ipt}$	1 if DSP suite has been retrofitted
A^{facility}_{it}	1 if suite has been built
I^U_{ipt}, I^D_{ipt}	USP/DSP inventory level
$I^{\text{under,U}}_{ipt}$	USP inventory amount deviating from strategic level
$I^{\text{under,D}}_{ipt}$	DSP inventory amount deviating from strategic level
K_{it}	1 if investment to construct facility i took place in period t
L^U_{ipt}	1 if retrofitting for product p in suite i starts at t
L^D_{ipt}	1 if retrofitting for product p in suite i starts at t
L_{ipt}, L_{jpt}	1 if licence payment for product p starts at t in suite i
Q_{ijpt}	flow of material from USP to DSP suite
S_{jpt}	sales amount of product p (units vary)
W_{jpt}	amount of product p which is wasted (units vary)
Y'_{ipy}	1 if new campaign starts
U^U_i, U^D_i	1 if there has been USP or DSP production in facility i
$Z^U_{ip'pt}, Z^D_{ip'pt}$	1 if there is a changeover from $p' \to p$ in USP or DSP suites
Δ_{pt}	demand not met

Free Variables

Cost	manufacturing cost (to be minimised)

Chapter 4

Biopharmaceutical Capacity Planning using a State Task Network Topology

4.1 Introduction

In the previous chapter a standard MILP formulation was developed to address the challenge of biopharmaceutical capacity planning involving batch and perfusion processes as well as build versus outsourcing production. Owing to the computational complexity of the model, a rolling time horizon was required in order to obtain solutions within reasonable time. This proved to be successful, but one of the issues of using a rolling time horizon is that it is highly unlikely that the overall optimal solution will be found, since at each sub-problem there is an optimality gap. A method which could potentially solve the problem without using a rolling time horizon would therefore be beneficial. This chapter explores a more efficient mathematical formulation for the problem involving a state task network representation. An alternative method using a genetic algorithm was also developed, but was unfortunately not as successful. Details of that algorithm can be found in Appendix A.

Resource task networks (RTNs) and state task networks (STNs) have been

described before in literature as a way of formulating a problem in a more generic manner. Kondili et al. (1993) were the first to present the use of an STN as a method of representing batch processes for short term scheduling problems. By modelling the problem as an STN, ambiguities surrounding a process input and output streams were removed. RTNs differ from STNs in the way a problem is represented, namely the resources in RTNs show no distinction between raw materials, equipment and utilities. The states in an STN represent material states (for example, raw material, intermediate product, or final product). Using an RTN or STN can simplify the modelling constraints, and thus help improve performance.

The work presented in this chapter demonstrates how the standard mathematical representation introduced in the previous chapter can be reformulated using an STN topology, and how this contributes to performance improvements. The model is then expanded to include additional features which help provide more realistic manufacturing schedules. For example, retrofitting downtime is now included whenever a suite is retrofitted for a new product. This makes it less likely that a suite will be retrofitted for many products, since the downtime would adversely affect capacity. Once an upstream suite has been retrofitted, only products utilising the same process mode may be manufactured, unless another retrofitting takes place. This more accurately represents real-life scenarios. To add flexibility, only retrofitting downtime is applied to downstream suites. CMO production has been altered to allow for the simultaneous manufacturing of multiple products. This effectively means that the model can contain multiple CMOs with little added computational effort. A constraint on the minimum amount of material outsourced to a CMO in a given year has been added, to more closely represent the nature of contract manufacturing. A feature of the STN presented in this work is the ability to change time resolutions. In addition to a rolling time horizon, a two month time resolution has been compared to the one month resolution to determine whether improvements could be made in performance and the solution.

An examination of the best way to expand in-house capacity through con-

struction is presented. A decentralised network of smaller facilities is compared to building one large facility, and the robustness of the solutions is tested under uncertain changeover times. Retrofitting options for existing facilities to accommodate different products as opposed to outsourcing capacity are examined. These different approaches of increasing manufacturing capacity each have trade-offs in terms of cost, time and risks. The model is finally used to de-bottleneck existing manufacturing schedules so as to improve strategic criteria and ascertain the likely impact on subsequent production.

The remainder of this chapter consists of an explanation of the problem domain followed by a description of the mathematical formulation used. An industrial case study is then used to explore the capabilities of the model, followed by a discussion of the results. The mathematical nomenclature can be found at the end of this chapter.

4.2 Problem Definition

The objective of this work is to minimise the total manufacturing cost of a long-term biopharmaceutical production plan, taking various constraints into consideration.

4.2.1 State-Task Network

This model has been formulated to closely follow state-task network representations. Figure 4.1 describes how a typical bioprocess can be modelled as an STN, with circles representing states, and rectangles representing tasks. Raw material is converted to an intermediate product via an upstream process (USP) involving a cell culture. The intermediate product is then purified in a downstream process (DSP). For perfusion-based cell culture, often the intermediate product is frozen and purified at another time. Fed-batch processes have their intermediate product purified immediately.

In a network of multiple USP and DSP suites, it is possible for the intermediate product to be transferred and purified in different DSP suites (Figure

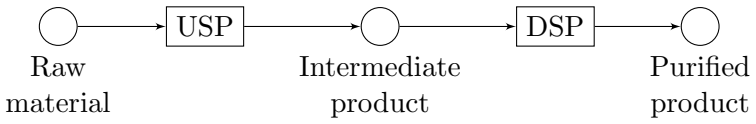


Figure 4.1: Process flow diagram using STN terminology

4.2). This is only possible for perfusion-based processes. The material states are duplicated for each USP/DSP suite since each suite has its own inventory levels and costs, thus must be modelled separately. With different facility capabilities, certain USP or DSP suites may not be utilised with particular products.

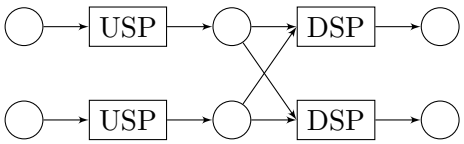


Figure 4.2: Basic network of suites with complete transfer capabilities

Each task takes a certain amount of time to convert material from one state to another, depending on the product and the task itself. Since the model uses discrete time intervals, the time required for each task is measured in time periods. The STN allows for easy manipulation of time resolution, whereby each time period can be changed from one month to two months if desired. This will reduce the size of the model, potentially improving performance. However, for perfusion based processes, which require a continuous and precise production time, altering the time resolution could have adverse effects on capacity. Therefore, the time resolution must be carefully chosen based on cell culture durations present in the case study.

4.2.2 Perfusion ramp-up times

When a perfusion process begins, there is a period of time where cell density increases until it reaches a steady state. This period is called ramp-up, and depending on manufacturing practices, the material obtained during this time is either used or discarded. This model assumes it is discarded. One complication that was not tackled in previous work is that the effective ramp-up time can vary depending on which product was being manufactured beforehand. This

characteristic only occurs if there are multiple bioreactors. When a new cell culture begins from scratch, each bioreactor sequentially undergoes ramp-up, and slowly all bioreactors will be online. However, if the same product was produced in the previous campaign, then it is not necessary that all bioreactors must be offline. Instead, a subset of the bioreactors undergo ramp-up at a time. In this way, the effective time that was lost due to ramp-up is less. Since sequence-dependent changeover variables are present in this model, it was possible to include this feature.

4.2.3 Retrofitting considerations

Previous work addressed retrofitting from the perspective of capital expenditure, but did not consider the implications of downtime associated with it. This model expands the retrofitting constraints such that there is downtime applied to both USP and DSP suites if they are retrofitted. To accomplish this, extra binary variables needed to be introduced. Once retrofitted, depending on bioreactor size and fittings within the suite, it may or may not be possible to continue production of products using a different cell culture process. This added complexity requires further constraints to be enforced. This model assumes it is possible to switch products freely in DSP suites, but not in USP suites, owing to the size of commercial bioreactors in fed-batch systems.

4.2.4 Contract manufacturing

The option to manufacture using a CMO is given more flexibility in this model, by allowing multiple products to be produced simultaneously. Originally, if three products were able to be manufactured in a CMO, they would have to be scheduled correctly with no overlap of production. Now, it would be possible for all three products to be manufactured at the same time, effectively creating three CMOs rather than one. There is still limited capacity in the CMO, but this added flexibility means extra CMOs can be used without increasing the number of variables or constraints in the model. A certain amount of time and money is required to start licensing agreements with a CMO. In addition, CMOs generally

require a minimum amount to be manufactured at commercial scale. Thus to ensure that this requirement is met, extra constraints are included to enforce minimum annual production levels.

4.2.5 Decentralised production

The question of whether to build one large facility or multiple smaller facilities is an important consideration when choosing how to expand capacity. Building one large facility can reduce the overall capital expense, but offers less flexibility in terms of scheduling. Additionally, the fact that the facility is located in one place can increase risk, since any unforeseen natural disasters could affect manufacturing capabilities.

A network of smaller facilities can spread any natural or geopolitical risk, and can also help minimise transportation costs to various markets across the world. The flexibility in scheduling of having multiple facilities also helps production teams find solutions when problems occur in one facility (for example, a contamination breach). The trade-off is that the overall costs may be higher when total fixed costs and capital expenditure are considered.

4.2.6 Multi-purpose facilities

Perfusion-mode cell cultures use vastly different bioreactor sizes than commercial scale fed-batch cell cultures. This is due to the increased productivity that perfusion processes allow, thereby reducing the required size of the bioreactors. In contrast, most antibody production is carried out using larger stainless steel bioreactors under fed-batch mode operation. Large stainless steel equipment is difficult to move, and thus increases the logistical challenge of switching between products. The dormant equipment must also be stored in a separate suite, which may not always be feasible. Therefore, a challenge exists to make a facility truly multi-purpose, whereby a production team can switch between different process modes if required.

If a decision is made that prevents both types of processes from being manufactured in the same suite, then there will be ramifications on capacity planning.

Instead, it may be elected that retrofitting a facility will allow production for all both types of process modes. This could be accomplished through careful planning and consideration of the retrofit, or through the use of alternative equipment (for example, the use of disposables). Therefore, it is important to analyse the requirements of a company based on their portfolio of products and facilities, to determine whether various retrofitting options could help with capacity planning and overall costs.

4.3 Mathematical Formulation

The following section describes the mathematics behind the model. The nomenclature can be found at the end of this chapter. Similar to the previous model, many of the variables have been duplicated for the upstream and downstream parts of the model (e.g., the number of batches produced). Therefore, the superscripts U or D denote upstream or downstream, respectively. This model continues to use a discrete time representation, with monthly time resolution (although a two month resolution is also examined in the results section). With a monthly resolution, an 8 year planning horizon would contain 96 time periods.

4.3.1 Technical and commercial constraints

Production Constraints

The number of upstream batches produced in time period t, for product p, in fermentation suite i, is denoted by B^U_{ipt}. Normally this could be calculated by multiplying the production rate by the time used in a particular time period. However, this requires extra time variables, and when taking into consideration perfusion campaigns, can lead to less tight constraints. The approach taken here utilises the idea of a state-task network (STN) to determine how many batches are produced in a given time period, regardless of whether the product is produced using a perfusion or fed-batch process, whilst still maintaining the correct modelling conditions. Each perfusion cell culture will have a duration in terms of time periods (τ_p). For each time period within a cell culture, a number

of batches are produced ($n^{\rho}_{ip\theta}$). The first part of Equation 4.1 sums over the cell culture duration in reverse order to find out where the current time period is within the cell culture, and then takes the number of batches from $n^{\rho}_{ip\theta}$. For example, if a cell culture begins in month one ($F_{ip,1} = 1$), and lasts 5 months, then if the number of batches for month 3 were to be calculated, the sum would expand to: $n^{\rho}_{ip,0}F_{ip,3} + n^{\rho}_{ip,1}F_{ip,2} + n^{\rho}_{ip,2}F_{ip,1} = n^{\rho}_{ip,2}$ (where $n^{\rho}_{ip\theta}$ is zero-indexed). From this quantity, a number of batches ($n^{U}_{ip'p}$) are subtracted due to changeovers ($Z^{U}_{ip'pt}$). This number that is subtracted will consider the downtime from cleaning the facility and moving equipment, as well as ramp-up times. In the previous model, ramp-up times were fixed per product, but in this model we allow greater flexibility by linking it to the sequence-dependent changeover variables. Note that the number of batches subtracted also depends on the facility being used, since the batch rates may be different, thus for the same amount of downtime, a greater or fewer number of batches will be subtracted.

$$B^{U}_{ipt} = \sum_{\theta=0}^{\tau_p - 1} (n^{\rho}_{ip\theta} F_{ip,t-\theta}) - \sum_{p' \in P_i} n^{U}_{ip'p} Z^{U}_{ip'pt} \qquad \forall\, p \in P_i \cap P^{\mathrm{p}} \cap P^{\mathrm{r}},\ t,\ i \in I_t \tag{4.1}$$

For fed-batch processes, the idea is the same, except that since the cell cultures are much shorter, rather than representing a single cell culture, F_{ipt} represents the fact that production is taking place in that time period. The value of $n^{\rho}_{ip\theta}$ is equal to the number of batches that can be produced in a time period, which could consist of multiple fed-batch cell cultures. For example, if an *E. coli* cell culture takes 5 days, then in a 30 day time period there could be a maximum of 6 batches (excluding downtime due to new campaigns and changeovers). Since the USP and DSP production in fed-batch processes are coupled to one another, there may be fewer than 6 batches if the downstream process is the bottleneck. In this case, one would be able to predetermine the actual number of batches that can be produced, and adjust $n^{\rho}_{ip\theta}$ accordingly. Also, since the fed-batch processes are not continuous (and thus do not need to run for the entire month),

the equality symbol has be changed to an inequality.

$$B^U_{ipt} \leq \sum_{\theta=0}^{\tau_p-1} (n^{\rho}_{ip\theta} F_{ip,t-\theta}) - \sum_{p' \in P_i} n^U_{ip'p} Z^U_{ip'pt} \qquad \forall\, p \in \{P_i - P^{\mathrm{p}} \cap P^{\mathrm{r}}\},\ t,\ i \in I_t \tag{4.2}$$

To ensure that only one cell culture can take place in any suite at any given time, Equation 4.3 is enforced. Not only does it prevent multiple products being produced in the same time period, but it also back-checks up to the cell culture duration for each product, blocking any future production. This makes the constraint more restrictive and helps fix the binary variable F_{ipt} in more places.

$$\sum_{p \in P_i} \sum_{\theta=0}^{\tau_p-1} F_{ip,t-\theta} \leq 1 \qquad \forall\, t,\ i \in I_t \cap I^{\mathrm{owned}} \tag{4.3}$$

To prevent the case where cell cultures begin near the end of the time horizon and there is not enough time for completion, the following constraint is introduced.

$$F_{ipt} = 0 \qquad \forall\, i,\ p, t \in T : t > (|T| - \tau_p + 1) \tag{4.4}$$

The calculation of downstream batches is similar to that of upstream, except the concept of cell culture duration does not exist. Thus, if production takes place in a time period ($Y^D_{ipt} = 1$), the number of batches is less than or equal to the maximum number of DSP batches, $n^{D,\max}_{jpt}$, minus any subtracted due to changeover downtime.

$$B^D_{jpt} \leq n^{D,\max}_{jpt} Y^D_{jpt} - \sum_{p' \in P_j} n^D_{jp'p} Z^D_{jp'pt} \qquad \forall\, t,\ p \in P^{\mathrm{r}},\ j \in J_t \cap J_p \cap J^{\mathrm{owned}} \tag{4.5}$$

There may be a minimum number of DSP batches that are required per production cycle. It should be noted that both $n^{D,\max}_{jpt}$ and $n^{D,\min}_{jpt}$ are parameters which can vary with time, hence allowing flexibility should there be planned downtime in one of the months, or if the maximum number of DSP batches changes due to

extra/fewer equipment.

$$B^D_{jpt} \geq n^{D,\min}_{jpt} Y^D_{jpt} \qquad \forall\, t,\, p,\, j \in J_t \cap J_p \tag{4.6}$$

For fed-batch processes, where the USP and DSP production are coupled to each other, the number of USP batches must equal the number of DSP batches, hence the following constraints:

$$B^D_{jpt} = B^U_{ipt} \qquad \forall\, t,\, p \in \{P^{\mathrm{r}} - P^{\mathrm{p}}\},\, j \in J_t \cap J_p,\, i = j \tag{4.7}$$

$$Y^D_{jpt} = F_{ipt} \qquad \forall\, t,\, p \in \{P^{\mathrm{r}} - P^{\mathrm{p}}\},\, j \in J_t \cap J_p,\, i = j \tag{4.8}$$

The amount of material required per DSP batch, x^{load}_p, must come from USP inventory, and the flow of material from a USP to DSP suite is represented as Q_{ijpt}. This only applies to perfusion-based products.

$$x^{\mathrm{load}}_p B^D_{jpt} = \sum_{i \in I_p \cap I_t} Q_{ijpt} \qquad \forall\, t,\, p \in P^{\mathrm{r}} \cap P^{\mathrm{p}},\, j \in J_t \cap J_p \tag{4.9}$$

To prevent more that one product being produced simultaneously in the same suite, the following constraint is included:

$$\sum_{p \in P_j \cap P^{\mathrm{r}}} Y^D_{jpt} \leq 1 \qquad \forall\, t,\, j \in J_t \cap J^{\mathrm{owned}} \tag{4.10}$$

Timing Constraints

Changeovers occur when there is a product switch within the same suite. The following equations enforce a changeover from product $p' \rightarrow p$ if p' was produced in the previous time period. Changeovers for upstream production need to be strict, because otherwise the model may choose to reduce the upstream production time by arbitrarily adding a changeover that should not exist. For perfusion processes, this can add inaccuracies to the modelling. If there is an idle period, this model will assume that the changeover will take place in the idle period and thus will not subtract from available production time. This is accomplished

by adjusting $n^{\delta}_{jp'p}$ accordingly. These changeover constraints are different to the ones used in the previous chapter, since they sum over one set of the products, thereby reducing the number of constraints. A pseudo-product is introduced to allow for idle periods, all real products are members of P^{r}. For distinct p' and p:

$$\sum_{p \in P_i} Z^{U}_{ip'pt} = F_{ip',t-\tau_{p'}} \qquad \forall\ p',\ t > \tau_{p'},\ i \in I_{p'} \cap I_t \cap I^{\mathrm{owned}} \tag{4.11}$$

$$\sum_{p' \in P_i} Z^{U}_{ip'pt} = F_{ipt} \qquad \forall\ t,\ p,\ i \in I_p \cap I_t \cap I^{\mathrm{owned}} \tag{4.12}$$

$$Z^{U}_{ip'pt} = F_{ipt} \qquad \forall\ p' \notin P^{\mathrm{r}},\ p,\ i \in I_p \cap I_t \cap I^{\mathrm{owned}},\ t = \tau^{\mathrm{start}}_i \tag{4.13}$$

Since changeovers are less strict for downstream production (owing to there not being any continuous downstream processes being modelled here), only one equation is necessary. By using the older version of the changeover constraint, the pseudo-product for downstream variables is not required, thereby reducing the number of additional discrete variables. The impact of the extra discrete variables would not warrant the reduction in the number of constraints.

$$Z^{D}_{jp'pt} \geq Y^{D}_{jpt} + Y^{D}_{jp',t-1} - 1 \qquad \forall\ t,\ p' \in P^{\mathrm{r}},\ p \in P^{\mathrm{r}},\ j \in J_{p'} \cap J_t \cap J_p \tag{4.14}$$

Availability Constraints

To prevent cases where production takes place in a suite which is not yet ready, the following constraints are included. A^{U}_{ipt} is equal to 1 if the suite is ready for production, and this is linked to whether the facility has been built (A_{it}) and/or retrofitted ($A^{\mathrm{retrofit},U}_{ipt}$). These variables are in turn linked to the investment constraints which follow this section.

$$F_{ipt} \leq A^{U}_{ipt} \qquad \forall\ t,\ i \in I_t,\ p \in P_i \cap P^{\mathrm{r}} \tag{4.15}$$

$$Y^{D}_{jpt} \leq A^{D}_{jpt} \qquad \forall\ t,\ j \in J_t,\ p \in P_j \cap P^{\mathrm{r}} \tag{4.16}$$

$$A^U_{ipt} \leq A_{it} \qquad \forall\, t,\ i \in I_t,\ p \in P_i \cap P^{\mathrm{r}} \tag{4.17}$$

$$A^U_{ipt} \leq A^{\mathrm{retrofit},U}_{ipt} \qquad \forall\, t,\ i \in I_t,\ p \in P_i \cap P^{\mathrm{r}} \tag{4.18}$$

$$A^D_{jpt} \leq A_{jt} \qquad \forall\, t,\ j \in J_t,\ p \in P_j \cap P^{\mathrm{r}} \tag{4.19}$$

$$A^D_{jpt} \leq A^{\mathrm{retrofit},D}_{jpt} \qquad \forall\, t,\ j \in J_t,\ p \in P_j \cap P^{\mathrm{r}} \tag{4.20}$$

Investment Constraints

In order to use a facility, it must first be built if it does not already exist. The following equation forces K_{it} to 1 if that facility is to be used. This pseudo-binary variable is then penalised in the objective function by a parameter representing the construction cost. The constraint also ensures that the facility is not available until after the construction time, τ_i^{build}.

$$A_{it} \leq A_{i,t-1} + K_{i,t-\tau_i^{\mathrm{build}}} \qquad \forall\, i,\ t > \tau_i^{\mathrm{build}} \tag{4.21}$$

Licence costs are also considered, and they are modelled such that the licence is per facility, rather than any individual suite.

$$A^U_{ipt} \leq A^U_{ip,t-1} + L_{ipt} \qquad \forall\, p \in P^{\mathrm{r}},\ i \in I_p,\ t \in T_i \tag{4.22}$$

$$A^D_{jpt} \leq A^D_{jp,t-1} + L_{jpt} \qquad \forall\, p \in P^{\mathrm{r}},\ j \in J_p,\ t \in T_j \tag{4.23}$$

Similar to the investment constraints in the previous chapter, these equations ensure that in order for a product to be manufactured in a facility, any relevant retrofitting must be carried out. For example, L^U_{ipt} is equal to 1 if retrofitting for product p starts at time t in facility i. The investment for retrofitting must be

spent τ_i^{retrofit} time periods before the facility becomes available for that product.

$$A_{ipt}^{\text{retrofit},U} \leq A_{ip,t-1}^{\text{retrofit},U} + L_{ip,t-\tau_i^{\text{retrofit}}}^{U} \qquad \forall\, p \in P^{\text{r}},\, i \in I_p,\, t > \tau_i^{\text{retrofit}} \quad (4.24)$$

$$A_{jpt}^{\text{retrofit},D} \leq A_{jp,t-1}^{\text{retrofit},D} + L_{jp,t-\tau_j^{\text{retrofit}}}^{D} \qquad \forall\, p \in P^{\text{r}},\, j \in J_p,\, t > \tau_j^{\text{retrofit}} \quad (4.25)$$

Retrofitting Constraints

One of the new features of this model is that of applying downtime when retrofitting takes place. When a facility is being retrofitted for another product, current production must stop. For these constraints to work, L_{ipt}^{U} and L_{jpt}^{D} must be made binary variables. The following equation forces there to be idle time in all time periods where retrofitting is taking place. Idle time occurs when $F_{ipt} = 1$ for the pseudo-product, $p \notin P^{\text{r}}$.

$$\sum_{\theta=0}^{\tau_i^{\text{retrofit}}-1} F_{ip,t+\theta} \geq \tau_i^{\text{retrofit}} \sum_{p' \in P^{\text{r}} \cap P_i} L_{ip't}^{U} \qquad \forall\, p \notin P^{\text{r}},\, i \in I^{\text{retrofit}},\, t \in T_i \quad (4.26)$$

A similar concept applies to retrofitting downstream production, except that the constraint here does not force idle time directly, but rather prevents other products from being manufactured. The reason for this is that, as mentioned in the downstream changeover constraints, there is no pseudo-product in downstream production. Without the pseudo-product, idle time cannot be directly enforced.

$$\sum_{\theta=0}^{\tau_j^{\text{retrofit}}-1} Y_{jp,t+\theta}^{\text{D}} \leq \tau_j^{\text{retrofit}} (1 - \sum_{p' \in P^{\text{r}} \cap P_j} L_{jp't}^{D}) \qquad \forall\, p \in P^{\text{r}},\, j \in J^{\text{retrofit}},\, t \in T_j \quad (4.27)$$

An upstream or downstream suite can only be retrofitted one product at a time, hence the following constraints ensure that this requirement is met for both

USP and DSP suites.

$$\sum_{p \in P^{\mathrm{r}} \cap P_i} \sum_{\theta=0}^{\tau_i^{\mathrm{retrofit}}-1} L^U_{ip,t+\theta} \leq 1 \qquad \forall\, i,\, t \in T_i \tag{4.28}$$

$$\sum_{p \in P^{\mathrm{r}} \cap P_j} \sum_{\theta=0}^{\tau_j^{\mathrm{retrofit}}-1} L^D_{jp,t+\theta} \leq 1 \qquad \forall\, j,\, t \in T_j \tag{4.29}$$

In the case of upstream suites, large equipment cannot be easily moved in and out, especially if they are fixed. In this model, it is assumed that if a USP suite has been retrofitted for a fed-batch product, then the perfusion processes can no longer take place, unless the suite is retrofitted again, thereby preventing fed-batch production. This limitation is only applied to USP suites, since it is assumed that the DSP suite is more flexible due to the nature of the equipment sizes.

$$\sum_{p' \in P^{\mathrm{r}} \cap P^{\mathrm{p}}} A^{\mathrm{retrofit},U}_{ip't} \leq 1 - L^U_{ipt} \qquad \forall\, p \in \{P^{\mathrm{r}} - P^{\mathrm{p}}\},\, i \in I^{\mathrm{retrofit}},\, t \in T_i \cap T_p \tag{4.30}$$

$$\sum_{p' \in \{P^{\mathrm{r}} - P^{\mathrm{p}}\}} A^{\mathrm{retrofit},U}_{ip't} \leq 1 - L^U_{ipt} \qquad \forall\, p \in P^{\mathrm{r}} \cap P^{\mathrm{p}},\, i \in I^{\mathrm{retrofit}},\, t \in T_i \cap T_p \tag{4.31}$$

CMO Constraints

This model places restrictions on the minimum amount of material that is produced in a CMO in a given year. This is to prevent the case where a CMO is used sparingly just to meet the demand on the infrequent occasion where in-house capacity is limited. It is more likely that, should a CMO be selected, there would be a minimum amount produced during that campaign. The following equation calculates how much material of a product was manufactured in a CMO that

year, C_{jpy}.

$$C_{jpy} = \sum_{t \in T_y \cap T_j} x^D_{jp} B^D_{jpt} \qquad \forall\, p \in P^{\mathrm{r}},\, j \in \{J_p - J^{\mathrm{owned}}\},\, y \tag{4.32}$$

If a CMO has been used in that year (X_{jpy}), then the minimum amount must be enforced:

$$C_{jpy} \geq x^{\mathrm{min,CMO}}_{py} X_{jpy} \qquad \forall\, p \in P^{\mathrm{r}},\, j \in \{J_p - J^{\mathrm{owned}}\},\, y \tag{4.33}$$

The following equation forces X_{jpy} to be equal to 1 when production of p has occurred in that year.

$$\sum_{y \in Y_t} X_{jpy} \geq Y^D_{jpy} \qquad \forall\, p \in P^{\mathrm{r}},\, j \in \{J_p - J^{\mathrm{owned}}\},\, t \in T_j \tag{4.34}$$

Utilisation Constraints

There are maximum utilization targets for in-house facilities, and thus constraints need to be put into place to accomplish this. For every in-house facility and each year, the following equations restrict the total time used for each product in each month of the year to be below the maximum allowed. Therefore, if the maximum desired facility utilization is 75%, $H^{\mathrm{available},y}$ can be set to 270 days. The model applies the same utilization target to both upstream and downstream suites. It should be noted that these constraints limit actual production time, rather than utilisation time. Therefore it does not include changeover or ramp-up times. These could be included if desired, or $H^{\mathrm{available},y}$ could be adjusted to reflect this difference.

$$\sum_{p \in P_i \cap P^{\mathrm{r}}} \sum_{t \in T_y \cap T_i} \frac{B^U_{ipt}}{r^U_{ip}} \leq H^{\mathrm{available},y} \quad \forall i \in I^{\mathrm{owned}}, y \tag{4.35}$$

$$\sum_{p \in P_j \cap P^{\mathrm{r}}} \sum_{t \in T_y \cap T_j} \frac{B^D_{jpt}}{r^D_{jp}} \leq H^{\mathrm{available},y} \quad \forall j \in J^{\mathrm{owned}}, y \tag{4.36}$$

The following two equations are shown if actual utilisation calculations are required. They were used instead of the two equations above when comparing to the previous chapter's model.

$$\sum_{p \in P_i \cap P^{\mathrm{r}}} \sum_{t \in T_y \cap T_i} (B^U_{ipt} + \sum_{p' \in I_p \cap P^{\mathrm{r}}} \alpha^U_{ip'p} Z^U_{ip'pt}) / r^U_{ip} \leq H^{\mathrm{available},y} \quad \forall i \in I^{\mathrm{owned}}, y$$

$$\sum_{p \in P_j \cap P^{\mathrm{r}}} \sum_{t \in T_y \cap T_j} (B^D_{jpt} + \sum_{p' \in J_p \cap P^{\mathrm{r}}} \alpha^D_{jp'p} Z^D_{jp'pt}) / r^D_{jp} \leq H^{\mathrm{available},y} \quad \forall j \in J^{\mathrm{owned}}, y$$

Fixed Cost Constraints

The modelling of fixed costs is the same here as it was in the previous chapter. Upstream and downstream suite use (U^U_i and U^D_j) is separated so that fixed costs can be attributed individually. These constraints check to see whether the suite has been used, and if so set U^U_i or U^D_j to 1. These variables are then used in the objective function when applying fixed costs. If a suite has never been used over the planning horizon (e.g., if it had never been built, or if no product was ever allocated to it), then no fixed costs need to be applied for that suite. Also note that only the facilities which are owned (I^{owned}) need to be subjected to fixed costs.

$$U^U_i \geq F_{ipt} \qquad \forall\, i \in I^{\mathrm{owned}},\, p \in P_i \cap P^{\mathrm{r}},\, t \in T_i \tag{4.37}$$

$$U^D_j \geq Y_{jpt} \qquad \forall\, j \in I^{\mathrm{owned}},\, p \in P_j \cap P^{\mathrm{r}},\, t \in T_j \tag{4.38}$$

Inventory Constraints

For USP production, the inventory level of product p in time period t in facility i is equal to the amount produced plus the inventory level in the previous time period, minus the amount of material transferred to purification suites. There may also be special inventory levels ($I_{ipt}^{U,\text{special}}$) which could represent initial inventory levels and/or injection of material from validation runs prior to commercial production. For DSP production, the idea is the same except that instead of transferring material to purification suites, the product is either sold or designated as waste. It should be noted that upstream inventory levels are only tracked for perfusion processes, since for fed-batch processes the material is immediately purified.

$$I_{ipt}^{U} = x_{ip}^{U} B_{ip,t-\tau_p^{\text{qc}}}^{U} + I_{ip,t-1}^{U} + I_{ipt}^{U,\text{special}} - \sum_{j \in J_p} Q_{ijpt} \qquad \forall\, p \in P^{\text{p}} \cap P^{\text{r}},\, i \in I_p \cap I_t,\, t \tag{4.39}$$

$$I_{jpt}^{D} = x_{jp}^{D} B_{jpt}^{D} + I_{jp,t-1}^{D} + I_{jpt}^{D,\text{special}} - S_{jpt} - W_{jpt} \qquad \forall\, p \in P^{\text{r}},\, j \in J_p \cap J_t,\, t \tag{4.40}$$

Strategic inventory levels are also incorporated by calculating any deviation from the target $I_{ipt}^{\min,U}$ and then penalising these deviations in the objective function.

$$I_{pt}^{\text{dev},U} \geq I_{pt}^{\min,U} - \sum_{i \in I_p} I_{ipt}^{U} \qquad \forall\, p \in P^{\text{p}} \cap P^{\text{r}},\, t \tag{4.41}$$

$$I_{pt}^{\text{dev},U} \geq \sum_{i \in I_p} I_{ipt}^{U} - I_{pt}^{\min,U} \qquad \forall\, p \in P^{\text{p}} \cap P^{\text{r}},\, t : t > D_p^{\text{last}} \tag{4.42}$$

$$I_{pt}^{\text{dev},D} \geq I_{pt}^{\min,D} - \sum_{j \in J_p} I_{jpt}^{D} \qquad \forall\, p \in P^{\text{r}},\, t \tag{4.43}$$

$$I_{pt}^{\text{dev},D} \geq \sum_{j \in J_p} I_{jpt}^{D} - I_{pt}^{\min,D} \qquad \forall\, p \in P^{\text{r}},\, t : t > D_p^{\text{last}} \tag{4.44}$$

Shelf-Life Constraints

The products have a limited shelf-life, and a constraint needs to be put in place to ensure that the product is sold before its lifetime expires, or else it should be discarded as waste. There is also a shelf-life for the intermediate product coming from upstream production. In this case, the product must be purified before it expires.

$$I^U_{ipt} \leq \sum_{j \in P_j} \sum_{\theta=t+1}^{t+\zeta^U_p} Q_{ijp\theta} \qquad \forall\, i,\, p \in P_i \cap P^{\mathrm{r}},\, t \in T_i : t < (|T| - \zeta^U_p) \tag{4.45}$$

$$I^D_{jpt} \leq \sum_{\theta=t+1}^{t+\zeta^D_p} S_{jp\theta} \qquad \forall\, j,\, p \in P_j \cap P^{\mathrm{r}},\, t \in T_j : t < (|T| - \zeta^D_p) \tag{4.46}$$

Sales Constraints

The amount sold is equal to the demand minus any backlogs. Since the backlogs can accumulate, the amount that was late in the previous time period is also considered, such that the model can make up for lost sales.

$$\sum_{j \in J_p \cap J_t} S_{jpt} = D_{pt} - \Delta_{pt} + \Delta_{p,t-1} \qquad \forall\, p,\, t \tag{4.47}$$

4.3.2 Objective function

The discount factor is calculated as:

$$\epsilon_t = \left(\frac{1+f}{1+g}\right)^{t-1} \tag{4.48}$$

where f and g are the inflation and interest rate respectively.

The individual costs have been broken down as follows:

$$\text{Inventory cost} = IC = \sum_i \sum_p \sum_t \epsilon_t \big(\rho_{ip}(I^U_{ipt} + I^D_{ipt}) + \rho^{\text{carry}}_{ip}(I^U_{ipt} + I^D_{ipt})\big) \tag{4.49}$$

$$\text{Inventory penalty cost} = IPC = \sum_{p}\sum_{t} \epsilon_t\big(I_{ip}^{\text{penalty}}(I_{ipt}^{\text{dev},U} + I_{ipt}^{\text{dev},D})\big) \quad (4.50)$$

$$\text{Variable cost} = VC = \sum_{i}\sum_{p}\sum_{t} \epsilon_t\big(\eta_{ip}^{\text{bias}}(\eta_{ip}^{U}B_{ipt}^{U} + \eta_{ip}^{D}B_{ipt}^{D})\big) \quad (4.51)$$

$$\text{Fixed cost} = FC = \sum_{i}\sum_{t\in T_i} \epsilon_t\big(u_i^{\text{cost},U}U_i^{U} + u_i^{\text{cost},D}U_i^{D}\big) \quad (4.52)$$

$$\text{Transportation cost} = TC = \sum_{i}\sum_{j}\sum_{p}\sum_{t} \epsilon_t(q_{ij}^{c}Q_{ijpt}) \quad (4.53)$$

$$\text{Waste cost} = WC = \sum_{i}\sum_{p}\sum_{t} \epsilon_t\big(w^{\text{cost}}W_{ipt}\big) \quad (4.54)$$

$$\text{Backlog penalty cost} = BPC = \sum_{p}\sum_{t} \epsilon_t\big(\delta_p\Delta_{pt})\big) \quad (4.55)$$

$$\text{Facility investment} = FI = \sum_{i}\sum_{t} \epsilon_t\big(\kappa_i K_{it}\big) \quad (4.56)$$

$$\text{Retrofitting cost} = RC = \sum_{i}\sum_{p}\sum_{t} \epsilon_t\big(\lambda_{ip}^{U}L_{ipt}^{U} + \lambda_{ip}^{D}L_{ipt}^{D}\big) \quad (4.57)$$

$$\text{Licence cost} = LC = \sum_{i}\sum_{p}\sum_{t} \epsilon_t\big(\lambda_{ip}L_{ipt}^{U}\big) \quad (4.58)$$

$$\text{Minimise Total Cost} = IC+IPC+VC+FC+TC+WC+BPC+FI+RC+LC \quad (4.59)$$

This chapter compares the STN to the standard representation (SR) described in the previous chapter. To make a fair comparison, not all the constraints listed above were used as described. The CMO features were excluded, as were the constraints under the retrofitting subheading. Inventory level calculations did not include special inventory levels, and inventory levels above strategic targets were not considered. Changeover times were included in the utilisation calculations.

Equations 1 - 4.59 form the STN* model, and represent the STN with new features.

Table 4.1: Process Data for Drugs in Case Study 2

	Product					
	p_1	p_2	p_3	p_4	p_5	p_6
Product type	Blood-factor	Blood-factor	Blood-factor	mAb	Fab	ADC
Process data						
USP						
Fermentation mode	Perf[a]	Perf[a]	Perf[a]	FB[b]	FB[b]	FB[b]
Cell culture duration (days)	150	120	60	10	3	10
Harvest (AU[c]/day)	120	130	490	-	-	-
QC/QA time (days)	60	30	30	-	-	-
DSP						
Lot size (AU[c])	320	450	1000	6	2	6
Duration (days)	1.5	1.5	1.5	1.5	3.5	3.5
Cost data						
USP						
Variable (RMU[d]/AU[c])	0.005	0.005	0.005	0.004	0.007	0.004
Fixed (RMU[d]/year)	65	65	65	3.5	6.6	3.5
DSP						
Variable (RMU[d]/AU[c])	0.002	0.002	0.002	0.02	0.02	0.1
Fixed (RMU[d]/year)	48	48	48	5	15	5
CMO (RMU[d]/AU[c])	-	-	-	0.23	0.56	1

[a] Perfusion [b] Fed-batch [c] Arbitrary units [d] Relative monetary units

4.4 Illustrative Example

In order to compare the STN to the standard mathematical model, optimisations were run for two different case studies. The first case study is the same one as presented in the previous chapter, consisting of four products and four facilities. The second case study consists of 6 products and 11 facilities. The second case study was chosen to be larger than the first case study, so that we could examine the performance of the mathematical model when presented with a larger portfolio of products and facilities. Product information regarding process data and costs are shown in Table 4.1. The six products are a mixture of perfusion mode cell cultures and fed-batch cell cultures, reflecting the different product types being manufactured (namely blood-factors and antibodies respectively). The fragment

Table 4.2: Product and facility capability matrix for Case Study 2. Note: B - both upstream and downstream capability, U - upstream only, N - no capability, * - retrofitting required.

Facility	Starting year	Product						USP scale	
		p_1	p_2	p_3	p_4	p_5	p_6	$p_1 - p_3$	$p_4 - p_6$
i_1	1	B	B*	B*	B*	B*	B*	6 x 200 L	2 x 2000 L
i_2	1	U	B	B	N	N	N	6 x 200 L	2 x 2000 L
i_3	1	U	N	N	N	N	N	2 x 200 L	-
i_4	5	N	U	U	N	N	N	6 x 200 L	-
i_5	7	N	B	B	N	N	N	6 x 200 L	-
i_6	3	N	N	N	B	B	B	-	2 x 2000 L
i_7	5	N	N	N	B	B	B	-	6 x 2000 L
i_8	6	N	N	N	B	B	B	-	2 x 2000 L
i_9	7	N	N	N	B	B	B	-	2 x 2000 L
i_{10}	8	N	N	N	B	B	B	-	2 x 2000 L
i_{11}	1	N	N	N	B	B	B	-	2 x 2000 L

Table 4.3: Costs for retrofitting, licences, and other start-up costs for Case Study 2 (relative monetary units)

		Product			
		p_1	p_2	p_3	$p_4 - p_6$
Licence /	i_1	5	5	5	5
start-up	$i_2 - i_3$	0	0	0	-
cost	i_4	-	5	5	-
	i_5	-	5	5	-
	$i_6 - i_{10}$	-	-	-	5
	i_{11}	-	-	-	7
USP retrofit	i_1	0	16.5	16.5	11
DSP retrofit	i_1	0	38.5	38.5	44

antigen binding (Fab) product uses *E. coli* as the cell culture expression system, and thus has a much shorter cell culture duration than mammalian cell alternatives. The downstream processing takes longer however, since extra steps are required for pegylation and further polishing.

Table 4.4: Demand and strategic inventory profiles for Case Study 2 (arbitrary units, $\times 10^2$)

		Year							
	Product	1	2	3	4	5	6	7	8
Demand	p_1	200	200	200	158	87	44	11	0
	p_2	0	0	51	121	151	178	211	220
	p_3	0	0	0	46	120	178	208	226
	p_4	0	0	0	6	11	16	30	30
	p_5	0	0	0	0	5	6	11	16
	p_6	0	0	0	0	3	5	6	11
USP	p_1	64	64	51	28	13	0	0	0
strategic	p_2	0	0	15	28	44	51	54	56
inventory	p_3	0	0	22	88	174	204	221	234
	p_4	0	0	0	0	0	0	0	0
	p_5	0	0	0	0	0	0	0	0
	p_6	0	0	0	0	0	0	0	0
DSP	p_1	100	100	79	44	23	7	0	0
strategic	p_2	0	0	33	56	89	107	110	116
inventory	p_3	0	0	12	46	89	104	114	120
	p_4	0	0	3	6	8	15	15	15
	p_5	0	0	0	3	3	6	8	8
	p_6	0	0	0	2	3	3	6	8

Similar to the first case study, not all products can be manufactured in all facilities. The manufacturing capability matrix and upstream bioreactor scale are shown in Table 4.2. Certain facilities need to be retrofitted in order to allow the manufacturing of products, and other facilities only have upstream capabilities. It should be noted that facilities $i_1 - i_3$ and i_6 are already existing in-house facilities, i_4, i_5 and $i_7 - i_{10}$ are future in-house facilities which need capital expenditure to be built, and i_{11} is a CMO. One of the features of the new STN model is that the CMO is seen as an option to manufacture using third parties, rather than

a single CMO facility. Therefore, multiple products can be manufactured using CMOs simultaneously, since different CMOs could be used if necessary. In the case that a facility is currently being used for production (clinical or commercial) of other products, or that a facility needs to be built before it can be used, there is also a starting year shown in the table, representing when a facility is available from.

Retrofitting costs and other costs associated with starting up production in a facility are shown in Table 4.3. As with Case Study 1, licence fees include biologic licence applications (BLA) and prescription user drug fees. There are also CMO negotiation costs, and any costs associated with technology transfer.

Demand and strategic inventory levels are shown for all products in Table 4.4. In this model, material from upstream production of antibody-based products is not stored, but instead processed immediately in a purification suite. Therefore, there are no USP strategic inventory levels for $p_4 - p_6$. Generally, the strategic levels are a function of demand in subsequent years, and are seen as a safety margin should unforeseen events, such as earthquakes, occur.

4.5 Results

The mathematical model from the previous chapter is referred to as the standard representation (SR), whereas the new formulation uses a state-task network (STN). There are also new features which have been included in the STN, but must be excluded whilst comparing the STN to the SR, otherwise infeasible solutions would occur during the comparison process. The STN with extra features is denoted by STN*. The optimizations were performed on an Intel Xeon W3565 Quad-core 3.2 GHz processor, with 6 GB RAM running Microsoft Windows 7 64-bit. The framework presented in this chapter uses the CPLEX 12.5.1 solver within GAMS 24.1.3.

Table 4.5: Model statistics for the SR and STN representations in Case Study 1 and 2. Case Study 2 also contains statistics for STN* and the 2 month resolution models.

	Model	Constraints	Continuous Variables	Discrete Variables
Case Study 1	SR	34,155	24,970	3,849
	STN	25,139	25,795	3,564
Case Study 2	SR	67,499	64,138	8,645
	STN	47,863	47,335	7,581
	STN 2 month	24,038	23,751	3,800
	STN*	45,928	41,437	8,733
	STN* 2 month	23,032	20,811	4,371

4.5.1 Model size

The purpose of reformulating the model as an STN was to increase performance and thereby obtain better solutions. All things being equal, the performance of a model can be improved upon by making the model smaller. In particular, fewer discrete variables and constraints can help increase computational efficiency.

The model sizes for the various mathematical representations for the two case studies are shown in Table 4.5. The STN shows a substantial reduction in the number of constraints, and a slight decrease in the number of discrete variables. This is largely down to improvements to changeover constraints, and no longer requiring constraints and variables for upstream inventory for fed-batch processes. The model assumes that material produced in a fed-batch process is immediately purified, and thus it makes no sense to keep track of upstream inventory levels. This is also applicable to strategic inventory levels. The slight reduction in the number of discrete variables is owing to the elimination of the Y^U_{ipt} variables that are present in the SR. The STN solely uses F_{ipt} to determine cell culture starting points, and negates the need for any further upstream product allocation variables. It should be noted that in order for the new changeover constraints to function correctly, a pseudo-product must be introduced. This pseudo-product allows for idle time, but also increases the number of discrete variables. Thus, the

STN should perform better as the number of products increases, thereby reducing the impact of the extra pseudo-product. This can be seen in the table, where the STN in case study 1 shows a 7.4% reduction in discrete variables, whereas in case study 2 the reduction is 12.3%, reflecting the fact that there are 6 rather than 4 products.

Owing to the way the STN is formulated, it is possible to change the time resolution. The resolution was changed to two months to see whether it could help improve performance. Having a resolution of two months means that each time period is 60 days rather than 30 days, and thus an 8 year capacity plan becomes 48 time periods rather than 96. This drastically reduces the size of the model, as shown in Table 4.5. The performance implications are discussed later. The STN with extra features (STN*) has a greater number of discrete variables, since the retrofitting variables L^{U}_{ipt} and L^{D}_{jpt} have been moved from being continuous to discrete. This is so that the retrofitting constraints work correctly for the downtime that is now applied when retrofitting takes place. Although there are now new constraints regarding retrofitting, there are fewer surrounding the scheduling of production in the CMO, thus the overall number of constraints is slightly reduced. Again, the two month resolution of STN* also shows a dramatic reduction in model size.

4.5.2 Performance comparison

The performance of the models are shown in Table 4.6. In addition to the objective value, the actual cost, and customer service level (CSL) are also shown. The actual cost excludes penalty costs arising from backlogs and strategic inventory deficits. It is shown so that it is easier to determine how or why a particular solution is better than another. For example, in case study 1 the SR and STN rolling time horizons have very similar objective values, and yet the actual cost of the STN 4/1 RH is 35 units less. This means that for a similar objective value, it has found a cheaper solution that can still manufacture enough material to minimise penalty costs to a similar degree at the SR 4/1 RH. Showing the actual cost can also help identify when a solution or case study is heavily influenced by

Table 4.6: Computational results for the SR and STN for the two case studies. Case Study 2 also includes results for the 2 month resolution model, and the STN with extra features (STN*). Optimality gaps for the rolling time horizons (RH) are for each sub-problem. Actual cost excludes penalty costs. Customer service level (CSL) is also shown.

	Model	Obj	Actual Cost	CSL (%)	Time (s)	Optimality (%)
	SR	3667	2939	100	10000	12.8
Case	STN	3574	2760	100	10000	8.4
Study 1	SR 4/1 RH	3528	2631	100	3455	3
	STN 4/1 RH	3533	2596	100	1362	3
	STN 4/1 RH 3,2	3515	2564	100	1651	3,2[a]
	SR	5230	3213	97.4	36000	9.4
	STN	5029	2996	97.7	650	5
Case	SR 4/1 RH	5045	3050	97.4	2440	5
Study 2	STN 2 month	5689	3006	96.9	196	5
	STN*	3158	2620	100	36000	13.7
	STN* 4/1 RH	3249	3044	100	6611	5[b]
	STN* 2 month	3157	2969	100	32958	5

[b] Last sub-problem was run to 2% optimality
[b] Last sub-problem timed-out after 3600 seconds at 7.1 % optimality

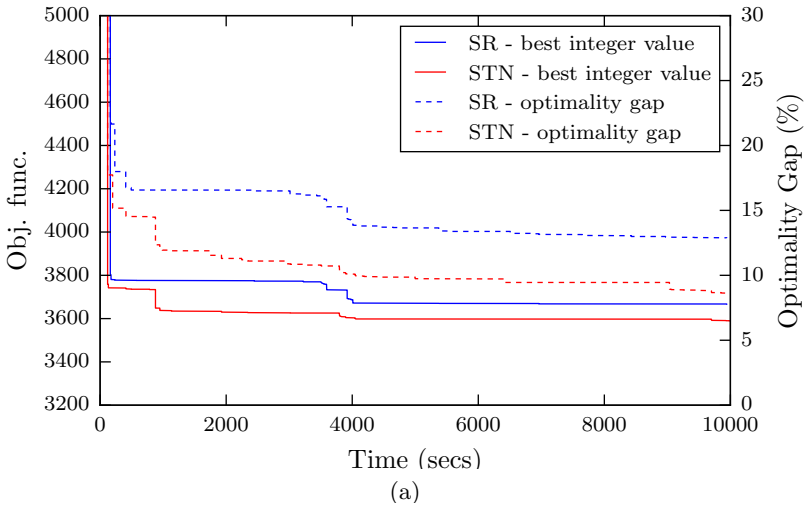
penalty costs.

Case study 1 proved to be problematic for both the SR and STN, but the STN has a tighter model and thus reached greater optimality. The STN also obtained a superior solution, both in terms of the objective value and the actual cost. A rolling time horizon was used to see whether better solutions could be achieved by breaking the problem into smaller sub-problems. Each sub-problem was optimised to 3% optimality, and as explained in the previous chapter, only optimises 4 years at any given point. The STN 4/1 RH is faster than SR 4/1 RH, but does not provide a better objective value. It was noticed that the STN was a tighter model, and the lower bound moved rapidly in the last sub-problem, not allowing enough time for superior solutions to arise. Therefore, the rolling horizon procedure was modified slightly so that the last sub-problem was optimised to 2% optimality, and this resulted in a better solution, with only a slight increase in computational effort. This modification was not applied to

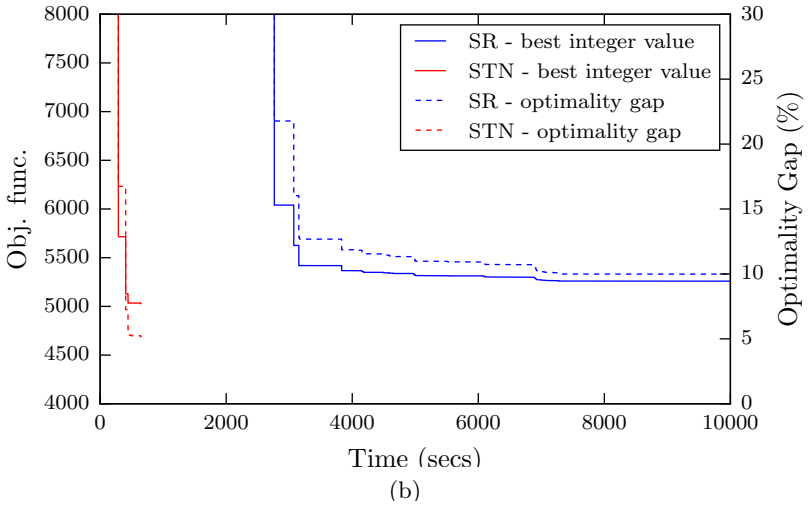
the SR 4/1 RH because it was already running substantially slower. Figure 4.3a shows the optimisation profile for the SR and STN, clearly demonstrating that at any given point in time, the STN provides not only a better solution but also greater optimality.

Case study 2 has a larger number of products and facilities, and therefore the models are greater in size (as seen in Table 4.5). It would be expected, therefore, that the performance of the models for this case study would be worse, but as seen in Table 4.6 it is actually better in terms of the optimality gap being achieved. This can be attributed to the fact that there are large penalty costs being applied to backlogs, as observed by the large discrepancy between actual costs and objective values. Since the penalty costs overshadow the other smaller costs, the optimality gap is closed faster. In case study 1, the CSL was 100%, and thus there were no backlogs. Case study 2 has approximately 97% CSL (excluding STN* which has new features which help alleviate capacity bottlenecks). Nevertheless, despite obtaining better optimality, the SR was still intractable, timing out after 10 hours of optimisation. The STN, however, reached 5% optimality in just over 10 minutes, obtaining a much better solution in the process. To improve upon the SR's performance, a rolling time horizon was used once again. Whilst the rolling time horizon did improve upon the SR (both in terms of objective value and time), it did not outperform the standalone STN. Figure 4.3b shows the performance profile for the SR and STN in case study 2, and it can be noticed that for a given solution, the STN provided a tighter lower bound and therefore greater optimality. The SR only slowly improved upon its solution and lower bound, and from 8000 seconds until 36,000 seconds (only the first 10,000 seconds are shown in the figure), the model barely improves the optimality or objective value. It should be noted that the solution from the STN was subsequently used to fix the optimisation for the SR to ensure that it was indeed a feasible solution for both models.

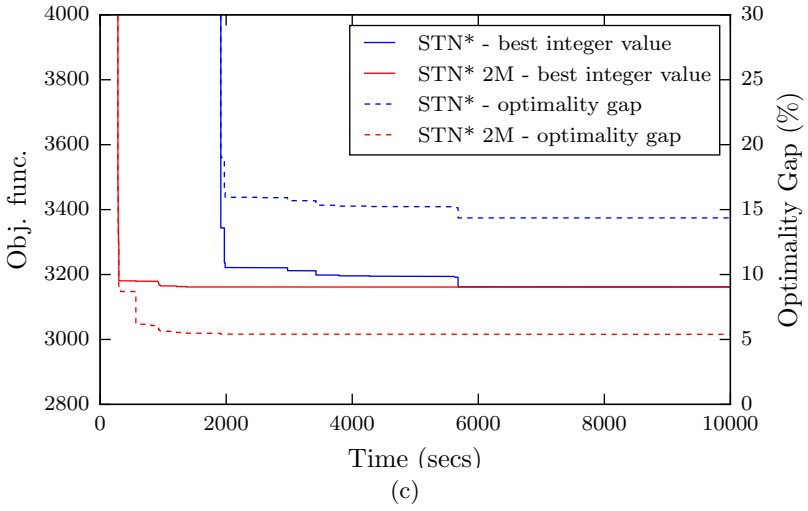
To see whether the STN's performance could be improved further, a two month resolution version was run. As explained previously, this would mean that the model would be approximately half the size, and thus would hopefully run



(a)



(b)



(c)

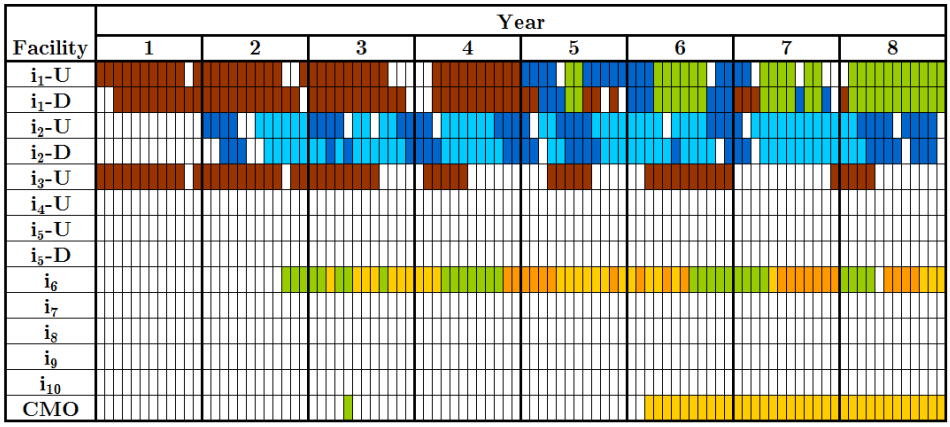
Figure 4.3: Optimisation profiles for (a) SR and STN for Case study 1, (b) SR and STN for Case study 2, and (c) STN* and STN* 2 month resolution for Case study 2

faster. Table 4.6 shows how the STN two month model does indeed run much faster, but at the expense of a worse solution. The issue with a two month resolution is that when capacity becomes limited, the model is less flexible, and thus cannot produce enough to meet demand or strategic inventory targets. Originally, the STN had a CSL of 97.7%, thus it was already clear that there were issues with backlogs. The two month resolution compounds this problem, resulting in a greater objective value. When penalty costs are removed, it can be seen that the actual costs for the STN and STN two month resolution are relatively similar, but the CSL has dropped for the two month resolution model. Allowing the optimisation to continue for longer, and thus achieve greater optimality, would not help in this situation. When comparing the two month resolution (2M) to the one month resolution (1M) model, the solution from 2M was converted offline so that it was compatible with 1M, and then 1M was fixed with that solution (apart from idle time) and optimised. This second optimisation is incredibly quick, and reaches 0.1% optimality within seconds, but it is necessary to ensure the solutions are feasible, and that the costs are calculated correctly.

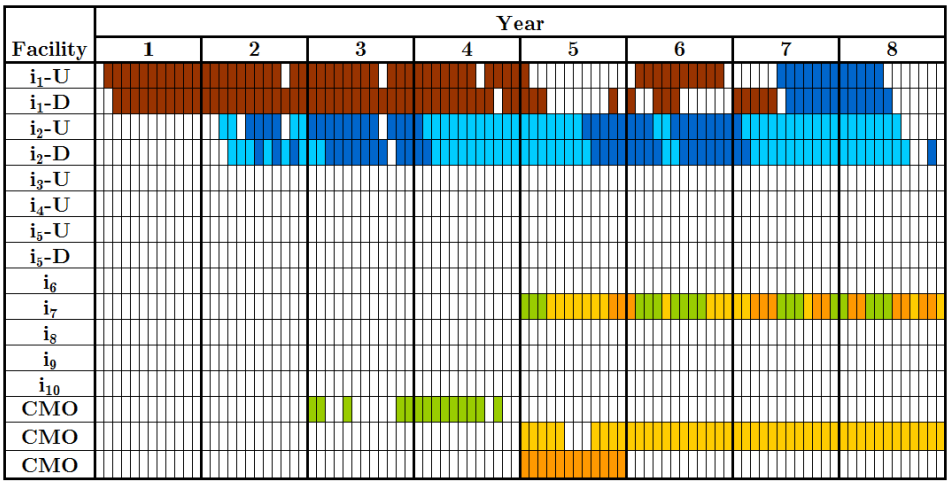
The second half of the results from case study 2 contain solutions from the STN with new features (STN*). One of the features which had a great impact on the objective value is that of initial inventory levels. Having initial inventory levels meant that the backlogs which occurred near the beginning of the capacity plan no longer existed, and therefore the penalty costs no longer applied. However, this did not mean that the optimisation was faster. Indeed, STN* only reached 13.7% optimality after 10 hours. This is consistent with case study 1, where the 100% CSL meant no backlog penalties, and greater optimality gaps. Whilst it could be argued that STN* had a greater number of discrete variables than the STN, this is unlikely to be the main cause of the performance bottleneck. It may be that without large penalty costs, the optimisation procedure finds it difficult to quickly differentiate between solutions, making it hard to tighten the lower bound and obtain better objective values. This is confirmed with the STN* 2 month resolution model (STN* 2M), which has a smaller model size than the original STN, and yet requires much longer to reach 5% optimality. A comparison

between the STN* and the STN* 2M can be seen in Figure 4.3c, and an important point to bear in mind is that although the STN* 2M required a long time to reach 5% optimality, the actual solution never changed from 1500 seconds onwards, and the optimality gap at that point was already approximately 5.5%. Thus is was a slow tightening of the lower bound which allowed it to reach termination point. It is a similar story for the STN*, which reached its best solution after almost 6000 seconds.

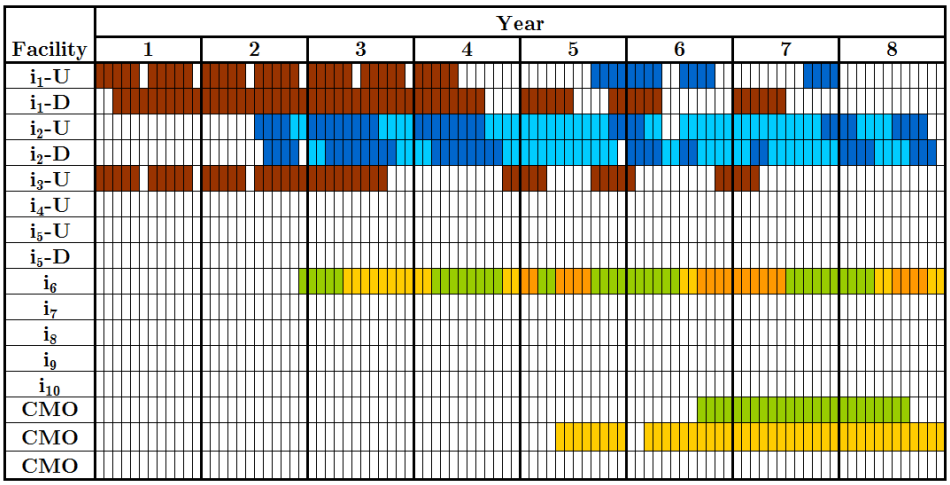
Interestingly, although the objective values are almost identical for the STN* and the STN* 2M, the STN* had a much lower actual cost (that is, the cost excluding penalty factors). Examining the Gantt charts (Figure 4.4) revealed that owing to the greater fine-grain control of production that the one month resolution has, STN* was able to eliminate the use of facility i_3, and therefore the fixed costs for i_3 were no longer being applied. The STN* 2M, however, did require the use of i_3. The product which was most problematic from a capacity perspective was p_1, and this product has a 150 day cell culture. With a one month resolution, this means 5 time periods are used for the cell culture. With a two month resolution, 3 time periods are required, each 2 months long. This means that each cell culture for this product requires an allocation of 6 months rather than the 5 months in the STN*, resulting in problems with backlogs. The STN* 2M combats this situation by using i_3, but this incurs a large fixed cost. The STN* 2M did however save on cost by using an existing facility i_6 for the production of antibodies, rather than build a new facility as per the STN*. Thus a hybrid between the STN* and the STN* 2M would provide the best solution. Whilst the actual cost of the STN* is lower, there were greater penalties attributed to strategic inventory levels, hence why the objective values between the one month and two month models are similar. A rolling time horizon was used with the STN*, but it did not achieve a better solution. This could be down to the fact that when optimising for the first 4 years, it chooses to use i_3 so as to minimise strategic inventory level penalties. The fixed cost is not as great, since it is only applied for 4 years at this stage. When expanding the optimisation to 5 years and above, the first years are fixed, thus i_3 will continue



(a) STN



(b) STN*



(c) STN* 2 month resolution

Figure 4.4: Comparison of Gantt charts for Case Study 2. Facilities i_6 - i_{10} and the CMO can only be used by fed-batch processes. Since these do not have upstream and downstream decoupling, it is not shown in the figures.

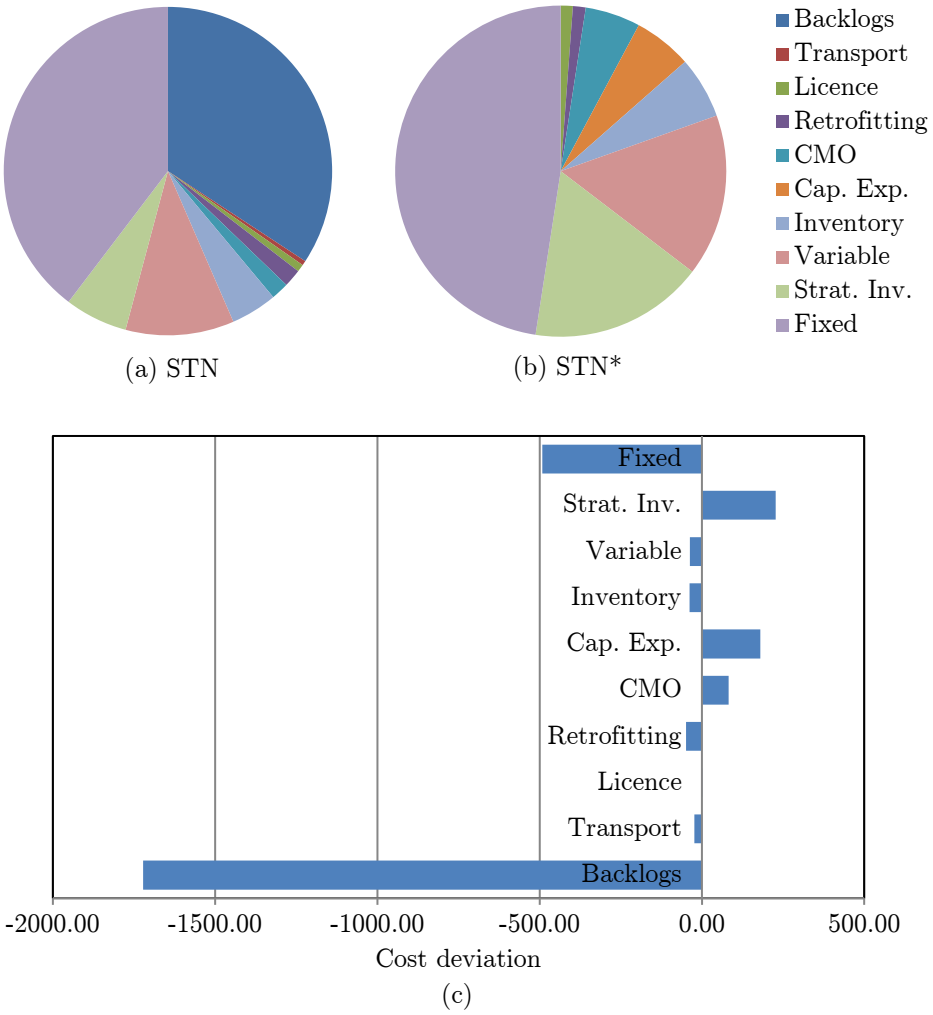


Figure 4.5: Cost breakdown between the STN and the STN with new features (STN*) in Case Study 2. Cost deviations are shown in (c), where positive values represent places where STN* has a greater cost.

to be used, but now the fixed costs associated with it will be extrapolated for a greater number of years, thus increasing overall cost.

4.5.3 Effect of new features on production planning

Comparing the Gantt charts of the STN* to the STN shows how the new features influence the solutions. First of all, initial inventory levels remove the need for facility i_3. Secondly, the retrofitting downtime constraints have meant that it is no longer optimal to start manufacturing p_4 in facility i_1. Instead, the demand for p_4 has been met in part by facility i_7 and a CMO. This is made possible by the STN*'s feature of being able to use a CMO for multiple products simultaneously.

In effect, it would be like having one CMO from years 3-8 (producing p_4 in years 3-4, and p_5 in years 5-8), and another CMO used just in year 5 for p_6. A breakdown of the costs between the STN and the STN* is shown in Figure 4.5. It confirms that the majority of the cost difference between the two models stems from backlog penalties in the STN. It also shows that fixed costs form the majority of the overall cost in both models, with the STN* having lower fixed costs in absolute terms for the reasons outlined earlier. The higher capital expenditure is owing to the construction of facility i_7.

Upon examination of the results, it seems that the model size is not a good indication of computational time required. It would appear that the penalty costs and overall spare capacity has a large impact on how quickly the optimality gap can be narrowed. It is not straightforward to determine what the penalty cost parameters should be, since if they are too small, the model will allow backlogs unnecessarily. If they are too large, the model solves very quickly, but not necessarily with the best real solution. Spare capacity (especially if there is symmetry between various manufacturing options) means it is more likely that penalties will have less impact (since demands can be met on time), and thus causes the optimisation to slow down as very similar solutions are analysed.

4.5.4 Decentralised manufacturing

The previous sections discussed the use of a state-task network design to improve performance. Furthermore, new features were included into the model to enhance the model's realism and accuracy. The subsequent sections continue to explore these new features and answer fundamental questions surrounding production planning.

One of the decisions that must be made when considering how to expand in-house capacity, is the kind of facility to be built. A company could choose to build either multiple smaller facilities or one large facility. The case study presented in this chapter has one large facility which contains 6 x 2000 L bioreactors, and two smaller facilities each containing 2 x 2000 L bioreactors. The downstream suites are sized per batch, and are therefore the same size between facilities.

For mammalian cell cultures, the upstream process is often the bottleneck, with durations of around 10-14 days. Downstream processes generally take far less time, and thus having multiple bioreactors function in parallel could greatly help eliminate the upstream bottleneck. The cost to build the large facility, i_7, is 180 RMU, whereas each small facility costs 90 RMU. In this section, the decision was made to keep production in-house, and hence the use of a CMO was disabled.

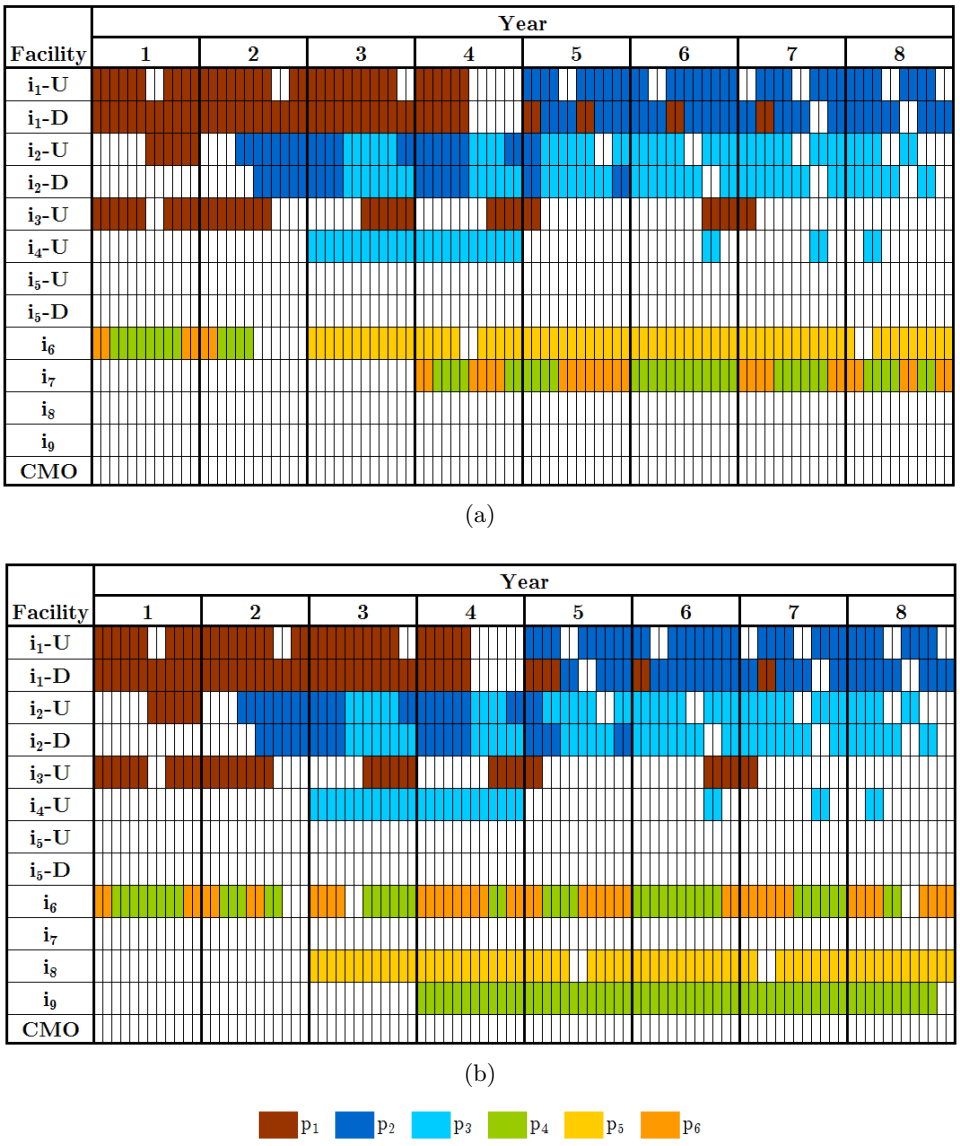


Figure 4.6: Effect of changeover times for fed-batch cultures on decision to build a centralised or decentralised capacity network. Changeover times are (a) 1 week and (b) 2 weeks. Facility i_7 is large, and $i_8 - i_9$ are small.

Figure 4.6 shows how the decision to build one large facility was chosen as the optimal solution in terms of cost with a standard changeover duration of

one week. The objective values were {Cost: 3526, CSL: 100%, Inventory: 84%}. The existing facility i_6 was retrofitted to allow for the production of p_5, and the new large facility i_7 was used to continue the production of p_4 and p_6. As soon as the changeover times for antibody production were increased to two weeks, the option of building a single large facility was unable to meet market demand unless facility i_1 was also retrofitted. Since retrofitting is expensive, the cheaper option now was to build two smaller facilities (Figure 4.6b). Utilising two smaller facilities allows for greater flexibility and fewer changeovers, hence it is able to cope with the increased changeover duration without any further retrofitting of i_1. In addition, since i_6 does not need to be retrofitted for the production of p_5, the other antibody products can continue to use the facility during year three, leading to higher inventory levels. The objective values for this solution were {Cost: 3576, CSL: 100%, Inventory: 88%}. The increased cost when compared to the large facility with one week changeovers is due to the fact that having two smaller facilities requires more fixed cost than one large facility, which accumulates over the 5-6 years that they are used.

Importantly, whilst the large facility had greater upstream capacity than the two smaller facilities combined, it was not able to cope with small changes in operating procedures in this particular case study. The more robust option was building two small facilities. The capital expenditure was the same, but in conjunction with i_6, three products could be manufactured simultaneously, rather than two in the large option. This greater flexibility increased the solution's robustness to uncertainty. If there were unplanned downtime due to a contamination or equipment failure, and one of the suites had to be closed down temporarily, having an extra suite would help satisfy demands. It should be noted that although not shown here, the scenario with the CMO re-enabled was also run. The findings were that the model still preferred using two smaller facilities (given two week changeover durations). The overall cost was slightly greater using a CMO, because the higher costs of producing in a CMO over 8 years outweighed the capital expenditure of building smaller facilities.

4.5.5 Retrofitting a multi-purpose facility

In the previous chapter, constraints were incorporated into the model so that when the upstream suite of facility i_1 was retrofitted for fed-batch production, the original perfusion-based products could no longer be manufactured. This was due to the fact that the 2000 L bioreactors used for the antibodies would be too large to move in and out of a suite. Additionally, if the equipment were moved from the suite, it would still need to be stored elsewhere. However, the impact of these operating restrictions can have huge consequences on capacity planning, as shown in Figure 4.7.

In this figure, it is assumed that the large facility i_7 was built, and that owing to an increase in changeover times, retrofitting i_1 was required. The option of using a CMO was removed, since there was the desire to keep production in-house.

When retrofitting for fed-batch processes blocks any further perfusion cell cultures, the manufacturing of products $p_2 - p_3$ must be conducted in facility i_5. This facility must be built, hence large capital expenditure must be employed. If, on the other hand, i_1 was truly multi-purpose and allowed both fed-batch and perfusion mode processes to be run, then the capital expense of building a new facility could be avoided. One may argue that a 30 day changeover time is not enough to move all the equipment. This is a valid concern, but can be addressed in two ways.

First, when retrofitting the facility, one could have in mind that the equipment cannot be moved, and thus enough space must be made available to house both sets of equipment. In this way, switching between products would not require extensive moving of large equipment, but extra cleaning and validation may be required to prove to regulatory bodies that good manufacturing practices are being maintained.

The second option is to use single-use bioreactors (SUB) for the fed-batch cell cultures. In this case study, the bioreactors are 2000 L in size, which is within the size limitations of SUBs. Using disposable equipment will negate the difficulty of moving large stainless steel bioreactors, and could allow the suite to become

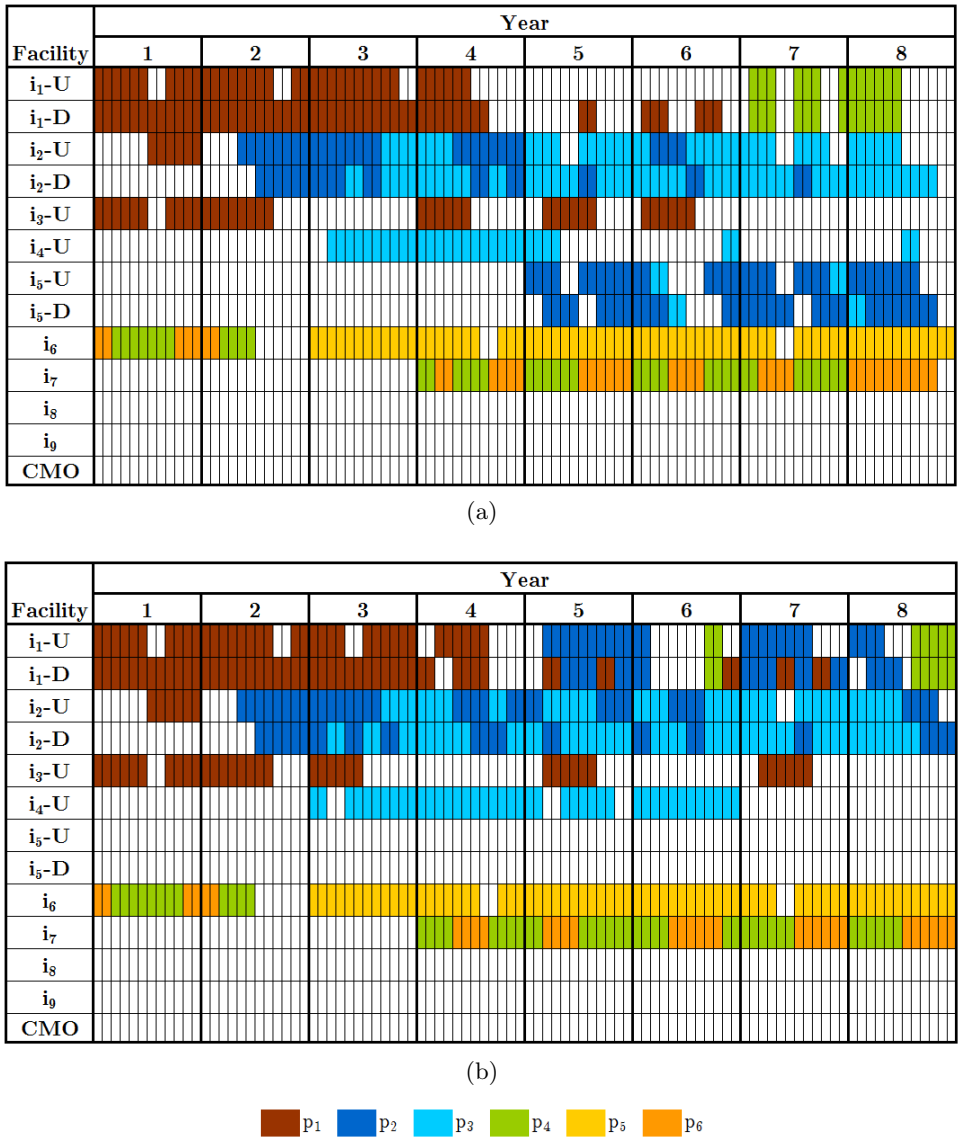
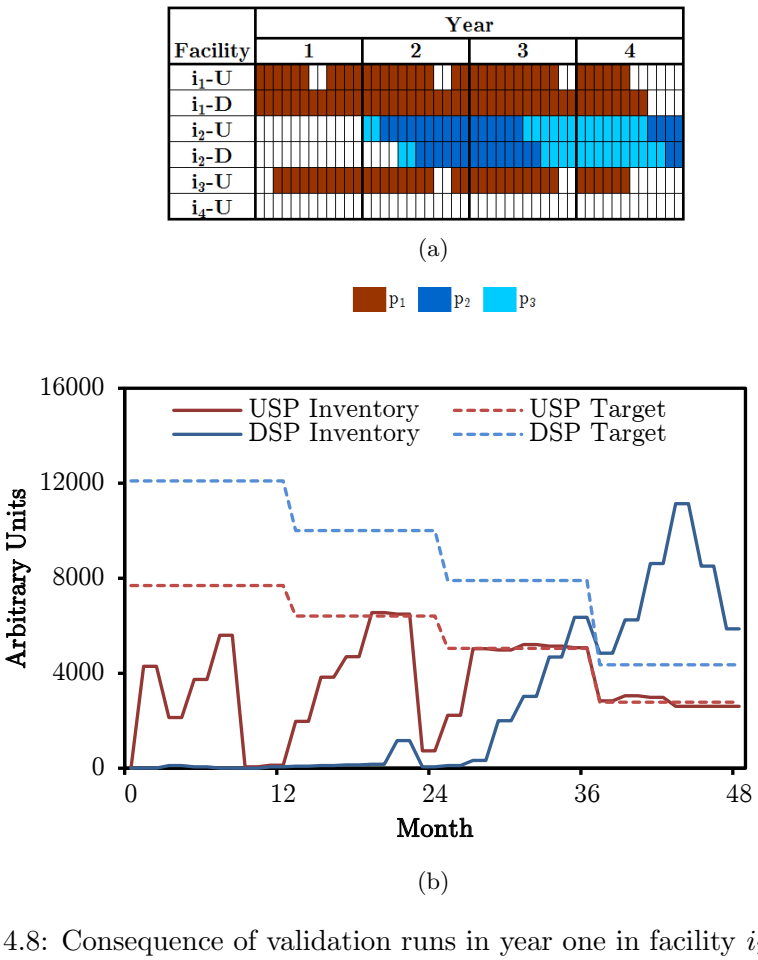


Figure 4.7: Multi-purpose facility options for i_1: (a) retrofitting for fed-batch processes restricts subsequent perfusion production, (b) fully multi-purpose facility with 30 day changeovers between different process modes.

multi-purpose.

The implications of restricting a suite to only one process mode can be large if it then forces the construction of new facilities. Building new facilities increases the risk a company adopts. Demand is only a forecast, and if the market demand reduces then the company will be left with an expensive, under-utilised facility. Retrofitting an existing facility eliminates much of this risk, and can also be performed quicker. The total costs for 4.7a and 4.7b are 3828 RMU and 3587 RMU respectively, highlighting the fact that in this case study, the better option

is to retrofit an existing facility to become multi-purpose.



(a)

(b)

Figure 4.8: Consequence of validation runs in year one in facility i_2 on p_1 inventory levels, (a) 4 year Gantt chart (b) inventory profile for p_1

4.5.6 De-bottlenecking production plans

Although it is important for the model to answer key questions regarding how and when to expand capacity, it is also useful to identify potential bottlenecks, and how they can be alleviated using current capacity restrictions.

In this case study, facility i_2 was originally being used for validation runs in year one for product p_2 and p_3 prior to starting commercial production. However, it was noted that the CSL for p_1 was predicted to fall below 100% near the end of year one as a result. At first glance, it would seem like this was being caused by a downstream bottleneck, since in year one there were two USP suites and only one DSP suite available for p_1. However, as shown in Figure 4.8b, the problem actually occurred when the USP inventory level reached 0 at around month 10.

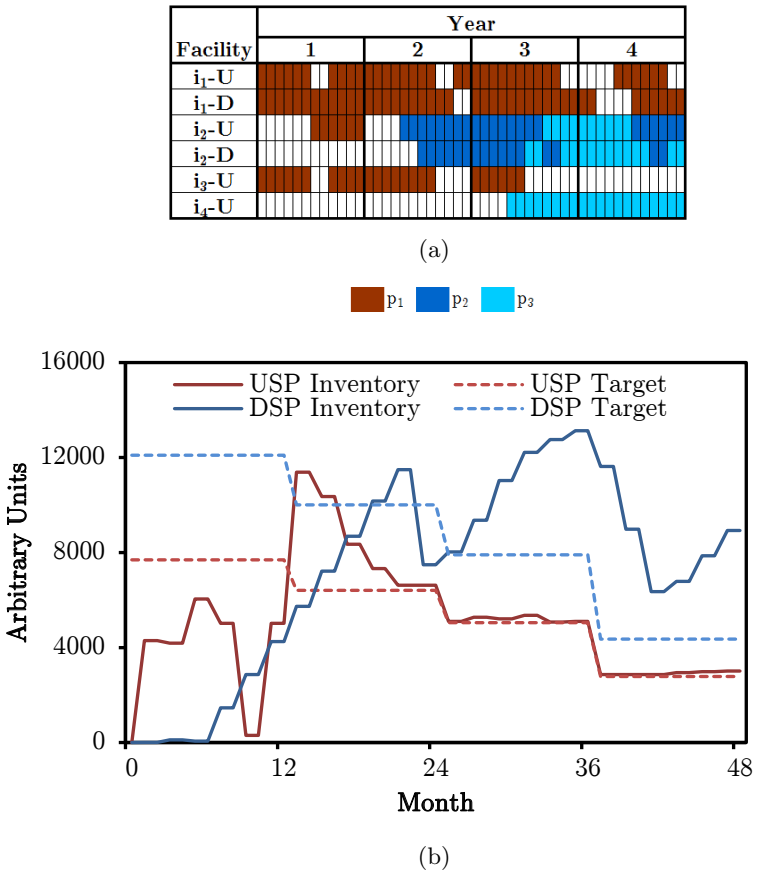


Figure 4.9: Effect of stopping the validation runs in i_2 6 months early, (a) 4 year Gantt chart (b) inventory profile for p_1

This suggested that although there were two USP suites available, it was actually the upstream process that was the bottleneck.

To analyse whether this could be circumvented by reinstating i_2 for USP production of p_1, the model was used to calculate the inventory profile and manufacturing schedule for the case where the validation runs were stopped after month 6 followed by one cell culture of p_1. Figure 4.9 shows the result of this change. It is evident that by adding just one campaign of p_1, the problem of USP inventory reaching 0 was eliminated, with the knock-on effect being that the downstream inventory level was much higher. This allowed the CSL to be maintained at 100%, and also reduced the risk during year two, where originally the DSP inventory level was very low. However, the validation runs in i_2 had to be continued in year two before commercial manufacturing could begin, hence there are 4 months of downtime at the start of year two (see Figure 4.9a). The

effect of this is that i_4 is required for the production of p_3, the demand for which could previously be met solely in i_2. Facility i_4 needs to be built, hence this change increases the capital expenditure of the company, albeit keeping the CSL at 100%. In reality, a company could choose to delay the launch of a product or reduce the amount it will supply to the market in the first few years, thereby removing the need to build a new facility, or at least delaying the construction. Here, it was assumed that the demands and launch date remained unchanged. Nevertheless, the model allowed the user to quickly determine whether a slight change to the production plan could help eliminate a bottleneck, and calculate the repercussions this would have on the production of other products in subsequent years.

4.6 Summary

This chapter has demonstrated the performance benefits of reformulating the mathematical model as a state-task network. The STN was able to outperform the SR in all cases, and in one of the case studies the standalone full-scale STN proved superior to the SR using a rolling time horizon. Issues still remain when the customer service level is close to 100%, and penalty costs have less impact on the objective value. A method which could eliminate the use of penalty costs may help with these issues, and could provide a more realistic production schedule.

The model presented here also demonstrated the use of extra features which make the scheduling more realistic from a production perspective. Retrofitting downtime has a large effect on determining optimal solutions. Allowing a CMO to be modelled as multiple third-parties rather than one facility, with minimum annual production limits, also adds more functionality and flexibility to the model.

In addition to the monthly resolution model, a two month time resolution was also used with the STN. Although faster, the solutions obtained from the two month model were not as good when capacity was nearing its limit. When capacity is not a large bottleneck, the two month model provides equally good solutions in much less time, reaching optimality targets in the process.

The advantages of using a decentralised production plan were discussed in the context of an industrial case study. The increased manufacturing flexibility of having multiple smaller facilities improved the robustness of the production plans should any unforeseen downtime occur.

The impact that different retrofitting options had on capacity planning was considered for multi-purpose facilities. The implications of preventing a facility from being truly multi-purpose, were an increase in capital expenditure, and greater risk associated with unused capacity in new builds.

Finally, the model was used to identify bottlenecks in production, and allowed the user to quickly determine how these bottlenecks could be removed. The impact this would have on the manufacturing of other products could then be analysed rapidly.

The next chapter discusses the use of a multi-objective model, which can help a production team determine which solution is best for them given their own strategic criteria.

4.7 Nomenclature

Indices

$i,\ j$	suite (alias)
$p,\ p'$	product (alias)
$t,\ \theta$	time period (alias))
y	year

Sets

I, J	suites
I_p, J_p	suites which produce product p
I_t, J_t	suites available in time period t
$I^{\text{owned}}, J^{\text{owned}}$	suites which are owned
$I^{\text{retrofit}}, J^{\text{retrofit}}$	suites which have retrofitting capabilities
P	products
P^{p}	products which are produced by perfusion
P^{r}	products which are real (not pseudo-product)
P_i, P_j	products which are produced by suite i or j
T	time periods
T_y	time periods which are part of year y
T_i, T_j	time periods which suite i or j are available in
Y_t	year which contains time period t

Scalars

H	time horizon (days)
$H_y^{\text{available}}$	maximum utilisation time in a year (days)
w^{cost}	cost to discard a unit of any product

Parameters

δ_p	backlog penalty cost
ϵ_t	discount factor
ζ_p^U	upstream product shelf-life (time periods)
ζ_p^D	downstream product shelf-life (time periods)
η_{ip}^U	upstream product batch cost
η_{jp}^D	downstream product batch cost
κ_i	facility investment cost
λ_{ip}	licence fees
λ_{ip}^U	retrofitting cost for upstream product
λ_{jp}^D	retrofitting cost for downstream product
ρ_{ip}	storage cost
ρ_{ip}^{carry}	carry of inventory cost
τ_p	perfusion cell culture duration (time periods)
$\tau_i^{\text{retrofit}}, \tau_j^{\text{retrofit}}$	retrofitting duration (time periods)
τ_i^{build}	suite construction duration (time periods)
τ_p^{qc}	time required for QCQA (time periods)
τ_i^{start}	starting time period for facility i
D_{pt}	demand of product p at time period t (units vary)
D_p^{last}	the time period containing the last demand of product p
I_p^{penalty}	penalty applied when strategic inventory is not met
$I_{ipt}^{U,\text{special}}$	USP additional inventory (e.g., initial levels)
$I_{jpt}^{D,\text{special}}$	DSP additional inventory (e.g., initial levels)
$I_{pt}^{\text{min,U}}$	USP strategic inventory level
$I_{pt}^{\text{min,D}}$	DSP strategic inventory level
$n_{ip\theta}^{\rho}$	number of USP batches that are produced in period θ of cell culture
$n_{ip'p}^{U}$	number of USP batches subtracted due to changeover time
$n_{jp'p}^{D}$	number of DSP batches subtracted due to changeover time
$n_{jpt}^{D,\text{max}}$	maximum number of DSP batches that can be produced in a time period
$n_{jpt}^{D,\text{min}}$	minimum number of DSP batches that should be produced in a time period
q_{ij}^{cost}	cost to transport intermediate material from facility i to j
r_{ip}^U	USP batch rate (batches/day)
r_{jp}^D	DSP batch rate (batches/day)
$u_i^{\text{cost,U}}$	USP fixed cost
$u_j^{\text{cost,D}}$	DSP fixed cost
x_p^{load}	downstream lot sizes (units vary)
$x_{py}^{\text{min,CMO}}$	minimum amount of p that should be produced in a CMO in a year
x_{ip}^U	USP batch output (units vary)
x_{jp}^D	DSP batch output (units vary)

Binary Variables

F_{ipt}	1 if a new cell culture of product p is started in facility i over period t
Y_{jpt}^D	1 if product p is produced in suite j over period t
L_{ipt}^U	1 if retrofitting for product p in suite i starts at t
L_{jpt}^D	1 if retrofitting for product p in suite j starts at t

Integer Variables

B^U_{ipt}	number of USP batches produced in suite i over time t of product p
B^D_{jpt}	number of DSP batches produced in suite j over time t of product p

Positive Variables

A^U_{ipt}, A^D_{jpt}	1 if USP or DSP suite is available
$A^{\text{retrofit,U}}_{ipt}$	1 if USP suite has been retrofitted
$A^{\text{retrofit,D}}_{jpt}$	1 if DSP suite has been retrofitted
A_{it}	1 if suite has been built
C_{jpy}	amount of product p produced in outsourced suites in year y
I^U_{ipt}, I^D_{jpt}	USP/DSP inventory level
$I^{\text{dev,U}}_{ipt}$	USP inventory amount deviating from strategic level
$I^{\text{dev,D}}_{jpt}$	DSP inventory amount deviating from strategic level
K_{it}	1 if investment to construct suite i took place in period t
L_{ipt}, L_{jpt}	1 if licence payment for product p starts at t in suite i or j
Q_{ijpt}	flow of material from USP to DSP suite
S_{jpt}	sales amount of product p (units vary)
W_{jpt}	amount of product p which is wasted (units vary)
U^U_i, U^D_j	1 if suite i or j have been used
X_{jpy}	1 if product p is produced in suite j in year y
$Z^U_{ip'pt}, Z^D_{jp'pt}$	1 if there is a changeover from $p' \rightarrow p$ in USP or DSP suites
Δ_{pt}	demand not met

Free Variables

Cost	manufacturing cost (to be minimised)

Chapter 5

Multi-Criteria Strategic Planning for Biopharmaceutical Production

5.1 Introduction

Biopharmaceutical companies face increasing pressure to meet market demands for multiple commercial therapeutics whilst minimising costs and capital expenditure. Different stakeholders will place emphasis on separate and often conflicting objectives, such as maximising customer service levels, posing various capacity planning challenges.

A tool that can incorporate various strategic criteria into the optimisation process is advantageous not only in terms of operational cost savings that result from optimal manufacturing schedules, but also in that it provides a clearer understanding of how uncertainty within the manufacturing environment can affect the robustness of a solution.

Hence this chapter builds upon the long-term production planning model described in the previous chapter by incorporating multiple objectives, including minimising the manufacturing cost whilst maintaining high customer service levels and strategic inventory targets. Two multi-objective methods are compared to

one another: goal programming and the ϵ-constraint method, and the advantages of using the ϵ-constraint method are discussed.

An industrial case study is presented with results showing how these factors, including varying the changeover times, can impact the different objectives and manufacturing schedules, highlighting some of the key challenges within strategic decision-making in the biopharmaceutical industry.

5.2 Problem Definition

The problem being discussed here is that of minimising the total cost of biopharmaceutical production over a finite time horizon. A diverse portfolio of different product types must be optimally allocated across a network of different facilities, each with their own manufacturing capabilities. The robustness of a solution is an important aspect to production planning, and thus methods are required to identify those solutions which would provide a more consistent schedule given various strategic criteria.

5.2.1 Multi-objective criteria

Often there are multiple conflicting objectives that must be met in biopharmaceutical production planning. For example, a company may wish to minimise the total cost and capital expenditure, while simultaneously satisfying market demands and maintaining strategic inventory levels. Whilst these concerns can be included in a single-objective model by applying penalty costs to any unmet criteria (for example, penalising backlogs), this method does not allow a user to easily determine how the different objectives interact with one another.

A multi-objective model can help a decision maker analyse the effects various objectives have on manufacturing decisions. In addition, by generating a set of optimal solutions one can determine which production plan is best suited to the company's attitude to risk.

5.3 Mathematical Formulation

This chapter adds a multi-objective component to the STN mathematical formulation outlined in the previous chapter. Three objectives were investigated: cost, customer service level, and strategic inventory levels. Two techniques were used to obtain multi-objective solutions, goal programming and the ϵ-constraint method.

5.3.1 Goal programming

The weighted goal programming method used here has been described in relation to biopharmaceutical capacity planning elsewhere (Lakhdar et al., 2007). One of the main reasons for using goal programming is that it is easy and intuitive to use for a decision maker, since placing targets on objectives and assigning weights is an understandable process. However, although easy to understand, sometimes it is not straightforward to determine *which* targets and weights to use, and this can have a large effect on the solutions found. There is also the issue of Pareto optimality, whereby one of the objectives in a solution provided by goal programming could be improved further without degrading any other objective. Although there are methods for fixing solutions so that they are Pareto optimal (Tamiz et al., 1999), there is still the problem of determining the full Pareto set of solutions.

Three key variables are introduced into the model. The goal target for each goal g, GT_g, the goal level, GL_g, and the deviations from the goal targets, $GD_{g,\text{dev}}$. There are two deviations that are possible, positive and negative, whereby both $GD_{g,\text{pos}}$ and $GD_{g,\text{neg}}$ are positive variables.

Cost

The total cost is equal to all operating costs and capital expenditure. More precisely, it includes inventory costs, variable costs, fixed costs, transport costs, waste costs, facility investment, retrofitting costs, and licence costs. The following equation sums these costs to form the goal level GL_{cost}. Individual components

of the equation are listed in the previous chapter.

$$GL_{\text{cost}} = IC + VC + FC + TC + WC + FI + RC + LC \tag{5.1}$$

Customer service level

The customer service level (CSL) is calculated as a percentage of demand that is met on time. The parameter P_t^d represents the number of products that have a demand in time period t, and is used to divide the summation of demand met, so that the CSL never exceeds 100%. $GLT_{\text{csl},t}$ represents the goal level for customer service level achieved for all products that have a demand in time period t. This variable is then summed over all time periods which have demands present, T^d, and divided by the cardinality of T^d (that is, the total number of time periods which have a demand present). This then forms the goal level for CSL, GL_{csl}.

$$GLT_{\text{csl},t} = 100 \times \sum_{p|D_{pt}>0} \frac{\left(\sum_{j \in J_p \cap J_t} S_{jpt}\right) - \Delta_{p,t-1}}{D_{pt}} / P_t^d \qquad \forall\, t \tag{5.2}$$

$$GL_{\text{csl}} = \frac{\sum_{t \in T^d} GLT_{\text{csl},t}}{\text{card}(T^d)} \tag{5.3}$$

Strategic inventory level

The inventory level for downstream production is often desired to be at a certain target level so that any problems with production have less of an impact on meeting demand. Here, the amount of inventory that is within the target level for a given time period is represented by $GLT_{\text{inv},t}$. The actual goal level, GL_{inv}, is then calculated by summing $GLT_{\text{inv},t}$ over all time periods which have an inventory target, T^{inv}. A value of 100% means that inventory targets have been

met for all products over all time periods.

$$GLT_{\text{inv},t} = 100 \times \left(\sum_{p|I_{pt}^{\min,\text{D}}>0} \left(1 - \frac{I_{pt}^{\text{dev,D}}}{I_{pt}^{\min,\text{D}}}\right)\right) / P_t^{\text{inv}} \qquad \forall\, t \tag{5.4}$$

$$GL_{\text{inv}} = \frac{\sum_{t \in T^{\text{inv}}} GLT_{\text{inv},t}}{\text{card}(T^{\text{inv}})} \tag{5.5}$$

Normalisation constraints

The difference between the goal targets, GT_g, and goal levels, GL_g, is used to calculate the deviation variables $GD_{g,\text{pos}}$ and $GD_{g,\text{neg}}$.

$$GL_g - GT_g = GD_{g,\text{pos}} - GD_{g,\text{neg}} \quad \forall g \tag{5.6}$$

In order to create an objective function which considers all goal deviations, they must first be normalised. The normalised deviations, $GD_{g,\text{dev}}^{\text{norm}}$ are calculated as follows:

$$GD_{g,\text{dev}}^{\text{norm}} = 100 \times \frac{GD_{g,\text{dev}}}{GT_g} \quad \forall g, \text{dev} \tag{5.7}$$

Finally, the weighted sum of these goal deviations is used in the objective function, where $w_{g,\text{dev}}$ represents the weight assigned to a particular goal deviation.

$$\text{minimise} \sum_{\text{g,dev}} w_{g,\text{dev}} GD_{g,\text{dev}}^{\text{norm}} \tag{5.8}$$

The full goal programming model consists of Equations 4.1-4.58, and 5.1 - 5.8.

5.3.2 ϵ-constraint method

One of the drawbacks of weighted goal programming is that of determining the weights and targets. Even once established, it is difficult to obtain a set of Pareto solutions using goal programming. Here, we describe the ϵ-constraint method,

and how it is applied to our model. The method used here is called AUGMECON2 and has been described in other work (Mavrotas and Florios, 2013).

The ϵ-constraint method works by converting all but one of the objectives into constraints, and then using an iterative process (see Figure 5.1) to find optimal values for all objectives. In general terms, to find the optimal solution $\mathbf{x}$ over p objective functions, $f_1(\mathbf{x})$ to $f_p(\mathbf{x})$, we use the following:

$$
\begin{aligned}
&\text{Problem P:} \\
&\max \left(f_1(\mathbf{x}) + eps \times (s_2/r_2 + 10^{-1} s_3/r_3 + \ldots + 10^{-(p-2)} s_p/r_p) \right) \\
&\text{st} \\
&\quad f_k(\mathbf{x}) - s_k = e_k \qquad k = 2 \ldots p \\
&\quad \mathbf{x} \in S
\end{aligned} \tag{5.9}
$$

where:

S is the feasible region of the original problem

s_k are non-negative slack variables

$e_k = \text{lb}_k + i_k \times \text{step}_k$

lb_k is the lower bound for objective k

$\text{step}_k = r_k/g_k$: the step for objective k

r_k is the range for objective k

g_k is the number of intervals for objective k

i_k is the counter for objective k for parametric variation of the constraint's RHS

eps is a very small number (10^{-3})

For this piece of work, the three objectives considered are cost, CSL and strategic inventory targets, represented by $f_1(\mathbf{x})$, $f_2(\mathbf{x})$ and $f_3(\mathbf{x})$ respectively. Since Equation 5.9 is formulated as a maximisation problem, the sign of the objective value for cost is inversed.

The method first creates a payoff table from which it determines a range of values for CSL and inventory targets. It splits these ranges into equally spaced

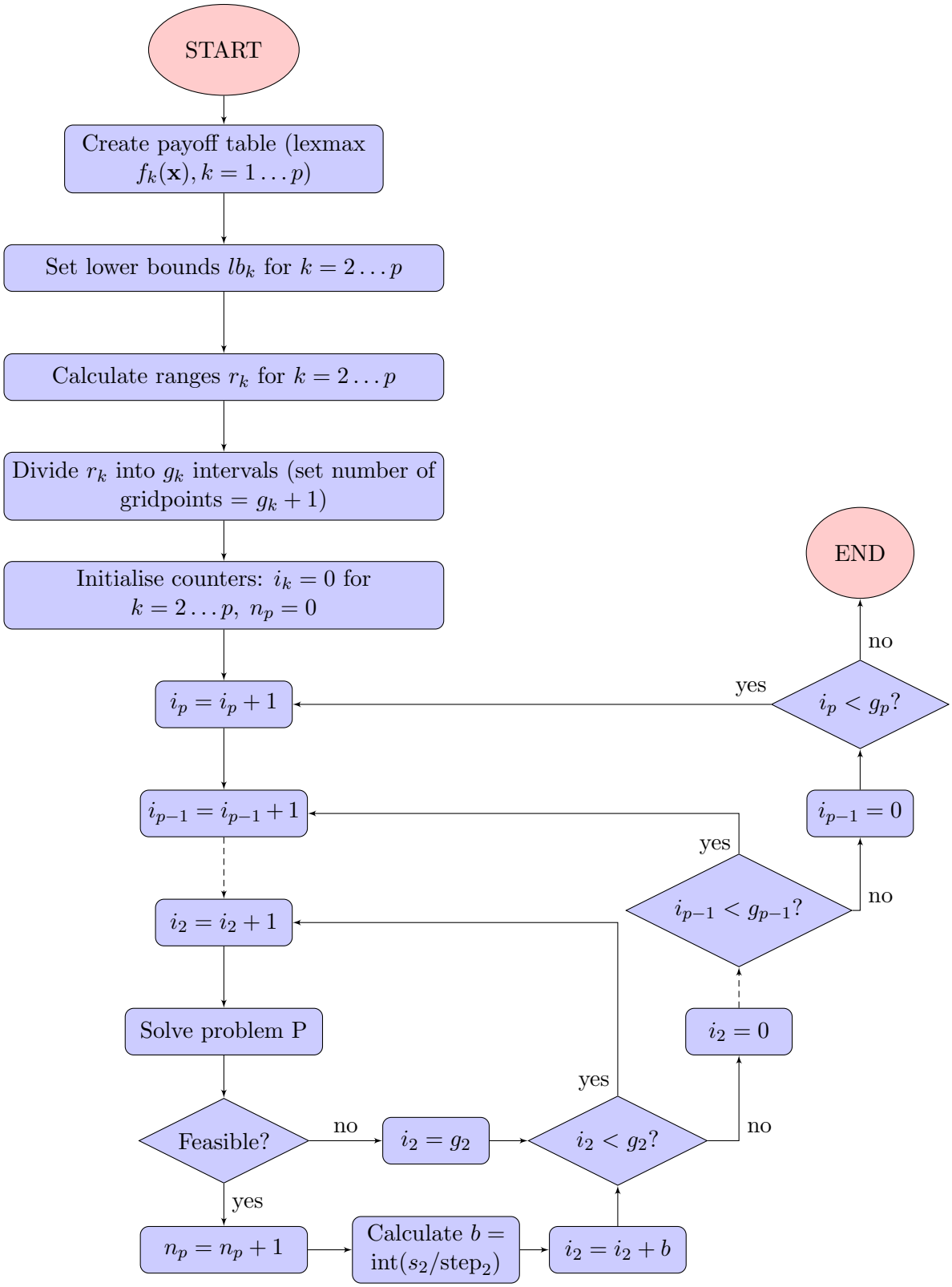


Figure 5.1: Flowchart of the ϵ-constraint method used in this work. Adapted from Mavrotas and Florios (2013)

points. It then picks the first point for inventory targets, and the first point for CSL. These points become the respective e_k values in Equation 5.9. The optimisation as per Equation 5.9 is run, whereby the lowest cost is found given that the two other objectives are at least as big as their e_k values. It then changes the e_k for CSL to the next point in its range, and repeats the optimisation. This is repeated until the CSL range is exhausted, at which point its e_k is reset to the first point, and the inventory target e_k is incremented to the next point in its range. The CSL e_k is again varied through its range. This process is repeated until e_k has been varied for all values in both ranges. This method, therefore, provides solutions for all combinations of CSL and inventory targets within a range and predefined number of grid points.

The individual objectives are calculated as per Equations 5.1-5.5. Equations 4.1-4.58 from the previous chapter form the remaining constraints. These equations, together with Equation 5.9, form the ϵ-constraint model.

5.4 Illustrative Example

The case study presented here consists of 6 products and 10 facilities. Product information regarding process data and costs are shown in Table 5.1. The costs for antibody products $p_4 - p_6$ were assimilated using discussions with industry and the commercial software BioSolve (Biopharm Services, Chesham, UK). Three types of antibodies were included: monoclonal antibodies (mAbs), fragment antigen binding (Fab) fragments, and antibody drug conjugates (ADCs). For in-house production, capital costs were removed from the calculations, since capital expenditure associated with equipment purchases and retrofitting were included separately in our model. Fixed costs in our model include labour, utilities, cleaning, insurance and taxes. The costs derived from consumables and materials constitute the variable costs. In order to calculate the costs for CMOs, capital charges were added to the previous costs, and the resulting cost was increased by 50% to reflect the additional expense of using a CMO. Costs for $p_1 - p_3$ were obtained from industrial discussion and used a variable:fixed cost ratio which

Table 5.1: Process data for products in Case Study 3

	Product					
	p_1	p_2	p_3	p_4	p_5	p_6
Product type	Blood-factor	Blood-factor	Blood-factor	mAb	Fab	ADC
Process data						
USP						
Fermentation mode	Perf[a]	Perf[a]	Perf[a]	FB[b]	FB[b]	FB[b]
Cell culture duration (days)	180	120	60	10	3	10
Harvest (AU[c]/day)	120	130	490	-	-	-
QC/QA time (days)	60	30	30	-	-	-
DSP						
Lot size (AU[c])	320	450	1000	6	2	6
Duration (days)	1.5	1.5	1.5	1.5	3.5	3.5
Cost data						
USP						
Variable (RMU[d]/AU[c])	0.019	0.019	0.019	0.004	0.002	0.004
Fixed (RMU[d]/year)	14.4	14.4	14.4	3.5	3.9	3.5
DSP						
Variable (RMU[d]/AU[c])	0.018	0.018	0.018	0.02	0.026	0.156
Fixed (RMU[d]/year)	9	9	9	4.7	7.1	5.6
CMO (RMU[d]/AU[c])	-	-	-	0.42	0.54	0.64

[a] Perfusion [b] Fed-batch [c] Arbitrary units [d] Relative monetary units

reflected the expected annual output.

The manufacturing capability matrix and upstream bioreactor scale are shown in Table 5.2. It should be noted that p_5 uses *E. coli* as its expression system, and thus it would be unlikely that it would be able to be produced in the same suite as a product using mammalian cells. Great effort would be required to show that there is a clear segregation between the processes, and that there is no sharing of equipment. The cost and risk involved in convincing regulatory bodies are likely to outweigh any benefit from having dual production, especially since the incentive of sharing equipment would be non-existent. With careful consideration, it may be possible to convince regulatory authorities by using disposable equipment, but this would probably be limited to toxicology or phase I

Table 5.2: Product and facility capability matrix for Case Study 3. Note: B - both upstream and downstream capability, U - upstream only, * - retrofitting required.

	Product						USP scale	
Facility	p_1	p_2	p_3	p_4	p_5	p_6	$p_1 - p_3$	$p_4 - p_6$
i_1	B	B*	B*	B*	B*	B*	6 x 200 L	2 x 2000 L
i_2	U	B	B	-	-	-	6 x 200 L	2 x 2000 L
i_3	U	-	-	-	-	-	2 x 200 L	-
i_4	-	U	U	-	-	-	2 x 200 L	-
i_5	-	B	B	-	-	-	6 x 200 L	-
i_6	-	-	-	B	B*	B	-	2 x 2000 L
i_7	-	-	-	B	-	B	-	6 x 2000 L
i_8	-	-	-	-	B	-	-	2 x 2000 L
i_9	-	-	-	B	-	B	-	2 x 2000 L
i_{10}	-	-	-	B	B	B	-	2 x 2000 L

Table 5.3: Costs for retrofitting, licences, and other start-up costs for Case Study 3 (relative monetary units)

		Product			
		p_1	p_2	p_3	$p_4 - p_6$
Licence /	i_1	5	5	5	5
start-up	$i_2 - i_3$	0	0	0	-
cost	i_4	-	5	5	-
	i_5	-	5	5	-
	$i_6 - i_9$	-	-	-	5
	i_{10}	-	-	-	7
USP	i_1	0	16.5	16.5	11
retrofit	i_6	-	-	-	8[a]
DSP	i_1	0	38.5	38.5	44
retrofit	i_6	-	-	-	6[a]

[a] Only applicable to p_5

Table 5.4: Capital expenditure required to build facilities, and their starting years.

	Facility									
	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}
Cost (RMU)	-	-	-	130	260	-	180	90	90	-
Starting year	1	1	1	3	4	1	4	3	4	2

Table 5.5: Demand and strategic inventory profiles for Case Study 3 (arbitrary units, $\times 10^2$)

		Year							
	Product	1	2	3	4	5	6	7	8
Demand	p_1	245	240	200	158	87	44	11	0
	p_2	0	0	101	202	253	253	240	220
	p_3	0	0	0	162	202	202	211	231
	p_4	0	8	10	18	20	28	30	35
	p_5	0	0	6	7	8	9	9	10
	p_6	0	3	3	6	6	9	11	12
USP	p_1	77	64	51	28	13	0	0	0
strategic	p_2	0	0	25	46	61	59	54	56
inventory	p_3	0	0	120	180	200	206	226	234
	p_4	0	0	0	0	0	0	0	0
	p_5	0	0	0	0	0	0	0	0
	p_6	0	0	0	0	0	0	0	0
DSP	p_1	121	100	79	44	23	7	0	0
strategic	p_2	0	0	50	95	128	122	110	116
inventory	p_3	0	0	41	76	101	106	115	120
	p_4	0	3	9	10	14	15	15	17
	p_5	0	0	4	4	5	5	5	5
	p_6	0	1	2	3	5	5	6	8

trials, rather than the commercial manufacturing being modelled here. Therefore, in this case study, if a facility is to be used for *E. coli* production, it must first be retrofitted. Mammalian cell cultures cannot continue after retrofitting has taken place. Of course, certain facilities can be built with just *E. coli* production in mind (i_8).

Retrofitting costs and other costs associated with starting up production in a

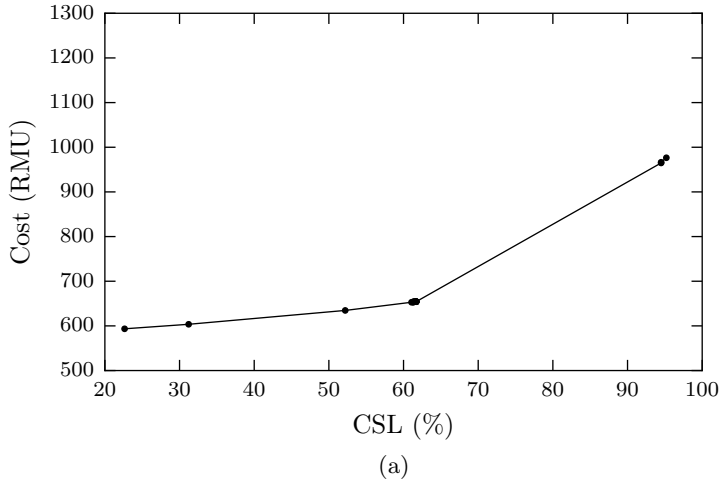
facility are shown in Table 5.3. The cost to build a facility, and the year that it can begin being used, is shown in Table 5.4. Demand and strategic inventory levels are shown for all products in Table 5.5. In this model, material from upstream production of antibody-based products is not stored, but instead processed immediately in a purification suite. Therefore, there are no USP strategic inventory levels for $p_4 - p_6$. Generally, the strategic levels are a function of demand in subsequent years, and are seen as a safety margin should unforeseen events, such as earthquakes, occur.

5.5 Results

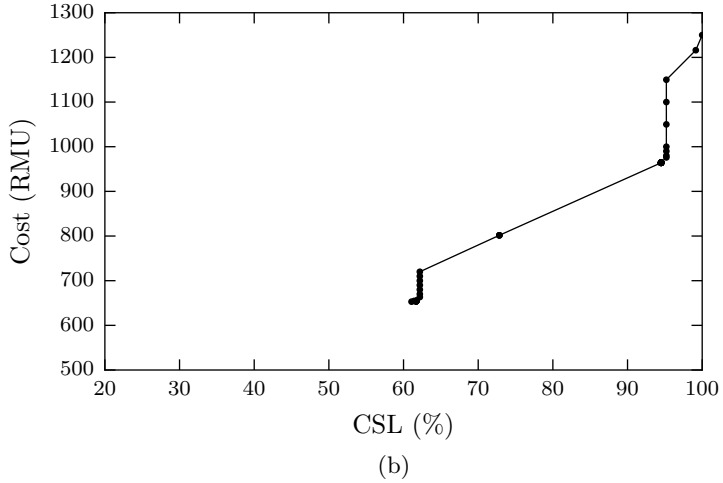
5.5.1 Comparing multi-objective methods

The first section of the results will discuss the merits of using the ϵ-constraint method over weighted goal programming for multi-objective problems. As mentioned earlier, one of the problems with using weighted goal programming is that it is difficult to obtain a complete set of Pareto solutions. Two ways in which a user could attempt to generate a set of solutions are (a) varying the weights, and (b) varying the targets. In contrast, the ϵ-constraint method eliminates the need of assigning weights or targets, and instead uses an iterative process to obtain a set of Pareto optimal solutions.

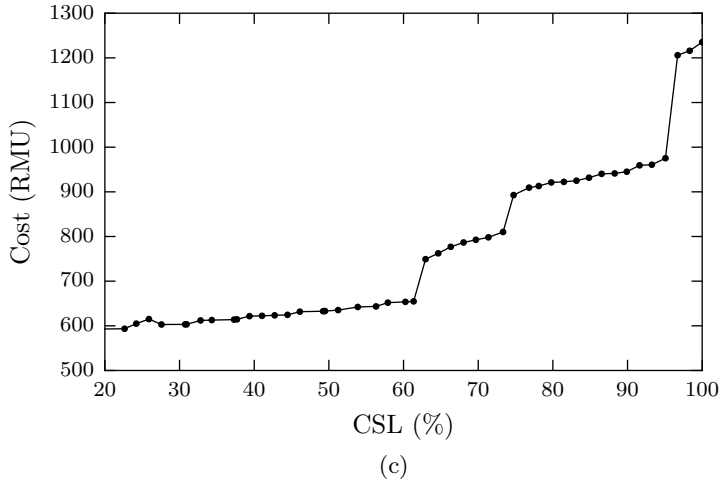
Figure 5.2 shows the different set of solutions generated for two objectives by goal programming methods and the ϵ-constraint method. The two objectives are total cost and customer service level (CSL). In Figure 5.2a, the cost and CSL target were kept constant (750 RMU and 100% respectively) and the weights were varied as follows. The weight for cost was kept at 1, and CSL was varied as 1.2, 1.4, 1.6, 1.8, 2, 3, ... , 8, 9. This was then repeated the other way round. Hence, in total 25 optimisations were conducted (24 from varying the weights, and one base case with the weights set at 1:1). Despite the fact that a wide range of weights were explored, the set of solutions obtained was not evenly dispersed. Instead, there are clusters of solutions around five distinct points. This shows two things: (a) the value of the weights chosen by the decision maker has a



(a)



(b)



(c)

Figure 5.2: Comparison of the sets of solutions obtained using (a) goal programming varying weights, (b) goal programming varying targets, and (c) the ϵ-constraint method.

real impact on the solutions obtained, and (b) varying the weights is not a good method of generating a Pareto set.

The effect of varying the targets on the set of solutions obtained was investigated next. Here, the weights were kept constant at 1:1. The CSL target was also kept constant, at 100%. The cost target was varied from 500 RMU to 1250 RMU, with steps of 10 RMU. Hence, 76 optimisations were conducted. As Figure 5.2b shows, whilst there is a slight improvement in the number of different solutions obtained, most are not useful since they are just dominated solutions. For example, at 95% CSL there is a set of solutions ranging from approximately 1000-1150 RMU. Obviously, for the same CSL, the solution with the lowest cost is preferable. This clearly illustrates the problem of Pareto inefficiencies generated using weighted sum goal programming. It should be noted that no attempt was made to repair the solutions to ensure Pareto optimality, since the added computational time required to accomplish this was deemed unnecessary given that other multi-objective techniques were available. In addition, even if solutions were repaired, it would be difficult to obtain an even distribution of solutions along the curve. The other issue visible in the figure is that no solutions with a CSL lower than approximately 60% were attained. This is due to the fact that the CSL target was kept constant at 100%, and highlights the problem of varying weights and targets to obtain a complete Pareto set.

The ϵ-constraint method used in this work should in theory be able to generate an evenly distributed set of Pareto optimal solutions. The CSL range used was 20-100%, with 48 evenly distributed optimisations. Figure 5.2c shows the resulting curve, and it is clear to see that the solutions obtained are superior to those from the goal programming method, since for any given CSL, the same or a lower cost is achieved. Owing to the nature of the method, the solutions are evenly distributed along the curve. However, Pareto optimality is only guaranteed if the individual optimisations are run to 100% optimality. Owing to computational reasons, this could not be achieved here. However, since the optimality gap used here was just 1%, the majority of the solutions generated were very close to being Pareto efficient, with only a couple of noticeable outliers (namely, around 25%

CSL).

The computational effort of obtaining the set of solutions presented in Figure 5.2 varies between methods. All three methods were executed for a three year capacity plan, with goal programming methods run to 3% optimality and the ϵ-constraint method to 1%. Changing the weights in goal programming resulted in a total time of 5310 seconds, changing the targets resulted in 3150 seconds, and the ϵ-constraint method required 2280 seconds. Given the fact that the ϵ-constraint method was not only quicker, but also provided a more evenly distributed set of solutions with greater Pareto efficiency, it seems to be a better choice when facing multi-objective problems like the one presented here.

5.5.2 Effect of variability on multi-objective criteria

The use of a multi-objective model is demonstrated through the heat maps presented in this section. Three objectives are considered: cost, customer service level, and strategic inventory levels. Using the ϵ-constraint method, the CSL and inventory level objectives are split into 8 points each, ranging from 70-100%. Thus, up to 64 optimisations are used to generate each heat map. Sometimes fewer than 64 optimisations are run, because the particular ϵ-constraint method used here can skip optimisations which provide no additional information. Here, the capacity plan is for three years and considered antibody production only. The colours on the heat maps represent total cost, which includes manufacturing costs and capital expenditure, amongst other costs (see Equation 5.1).

The changeover times between products are sometimes not known precisely until personnel have had the experience of manufacturing the products for themselves. Different operating practices and personnel experience levels can have an effect on the changeover times. Equipment sizes can vary widely between products and will also have an impact on the time, to the point where it may even be deemed infeasible to switch between certain products. In addition, if the seed train is to be produced in the same suite as the production-scale bioreactor, then the changeover times could easily be extended to 3-4 weeks for a mammalian cell culture at a 2000 L commercial scale.

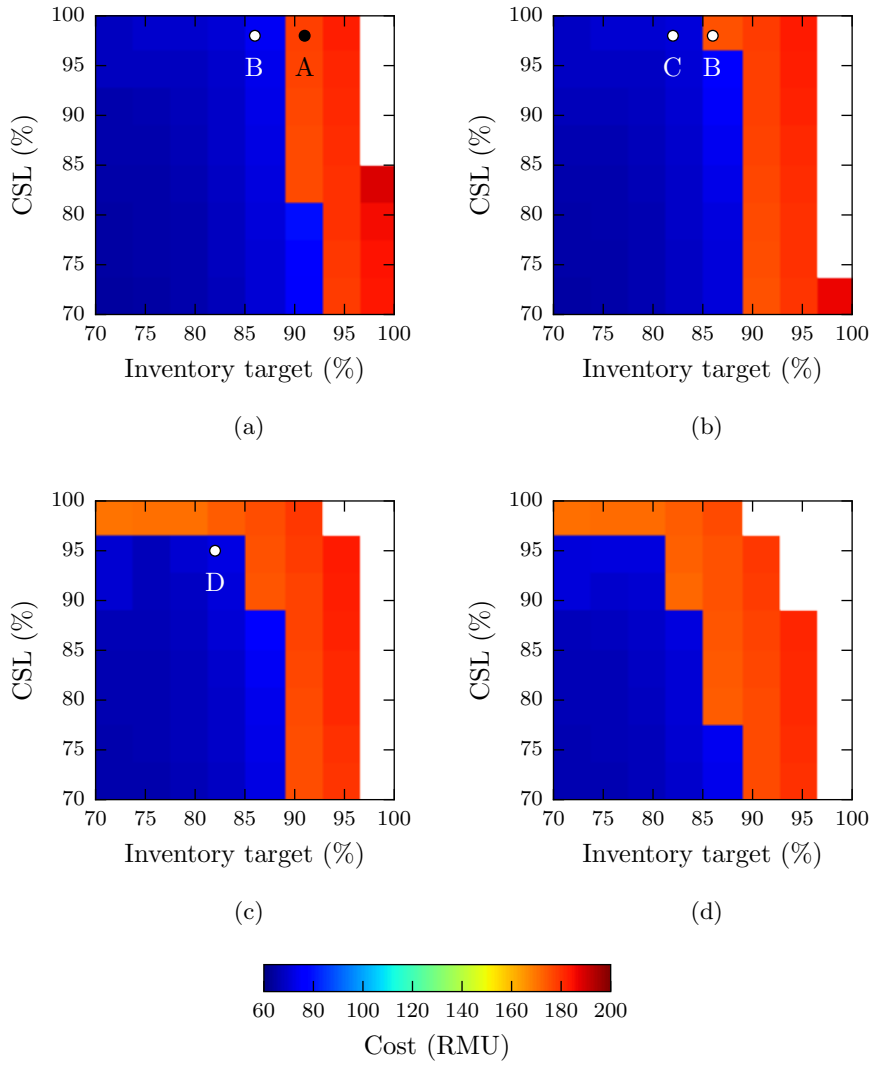


Figure 5.3: Heat maps illustrating the effect of changeover times on cost, customer service level and strategic inventory for antibody production in the first three years, without CMO availability. Changeover times are (a) 1 week, (b) 2 weeks, (c) 3 weeks, and (d) 4 weeks

Figure 5.3a shows design space when the changeover time is one week, and it is clear that there are three distinct areas on the heat map. The blue region (where cost is low) represents the design space where capital expenditure to increase capacity is not required. The red region is where a new facility needs to be built in order to meet inventory targets and CSL. There is also a white region, representing infeasibility given the current capacity options. Thus, with a one week changeover time, it is not possible to have both 100% CSL and 100% inventory targets met. It also shows how, depending on a company's risk

tolerance, one could drastically reduce cost by moving from point A to point B. The customer service level remains constant between the points, but the inventory target is reduced by 5%. This will negate the need to build a new facility and hence reduce capital expenditure by 90 RMU.

At first glance this seems to be a good trade-off, and worth the risk. However, if changeover times were to be extended to two weeks (Figure 5.3b), point B can no longer be maintained without a new facility. If the decision had already been made to forgo the new build and begin manufacturing at operating point B (with one week changeovers in mind), then it may be impossible to continue operating at point B altogether if changeovers became two weeks. This is because there is a construction time associated with building a facility. Thus a delay in the construction starting time would alter the design space. However, CSL can still be kept at 100% by moving to point C, at the expense of reducing inventory levels even further.

Matters are compounded when the changeover duration is extended to three weeks (Figure 5.3c). In this case, 100% CSL is not possible unless a new facility is built, thus if the original strategy was to operate at point B (which is no longer possible), then the CSL would have to drop to 95%, seen at point D. Of course, it may also be possible to reduce inventory targets to below 70% and achieve 100% CSL, but here we are assuming that management want to operate within the design space shown. Finally, four week changeovers are shown in Figure 5.3d. As can be seen from the heat maps, there is a gradual increase in the infeasible space as changeover times increase. However, so long as a new facility is built, there will always be the possibility to maintain 100% CSL. Thus, the less risky option, and potentially better option overall, would be to build a new facility, and thereby have a more robust manufacturing schedule that can cope with unexpected downtime or operating changes. It should be noted that these results disable the use of a CMO. The effect of using a CMO is discussed next, in the context of titre variation.

Titres often fluctuate throughout the course of commercial manufacturing, owing to the inherent uncertainty present in biological systems (Stonier et al.,

2013). There is also the possibility that titres for new products do not meet expectations as the process is scaled up to commercial manufacturing. As mentioned in Chapter 3, there is also the chance that a lower titre cell line is chosen if it generates fewer host cell impurities and thus allows for easier purification. Therefore, an analysis on the effect of a $\pm 20\%$ variation in titre for antibody products is examined in Figure 5.4. The antibody products were chosen because the case study used generic platform processes for their production, and thus key variables such as titre may change. The changeover time was kept constant at one week.

When the titre was reduced by 20%, it became impossible to satisfy demand and maintain 70% inventory targets using existing facilities (Figure 5.4a). A new facility had to be built (represented by the red region), but even then it may be risky should changeover times increase or other factors occur which further reduce capacity. The base case shown in Figure 5.4b is the same as that shown in Figure 5.3a. When titres increased by 20% in Figure 5.4c, the infeasible demands in the base case can now be met with the use of the new facility. The stark difference between the three figures show the large impact titres have on capacity requirements and the decisions that must be made accordingly. In Figure 5.3, it seemed like building a new facility would be sufficient in terms of meeting demand given uncertain changeover times. When variable titres are considered however, it may be that it is deemed too risky.

To reduce some of the inherent risk associated with the uncertainty surrounding titres and changeover times, one may wish to outsource capacity to a CMO. Figures 5.4d-5.4f demonstrate how by using a CMO one can satisfy demands and strategic inventory levels at all times. Notice that the dark blue regions cover the same areas as those covered in the respective 'non-CMO' heat maps, highlighting the fact that the model will use in-house facilities where possible, and only outsource when using a CMO is cheaper than any other alternative (for example, building a facility). For this particular case study, it would seem that a CMO is the least risky option, and also offers a reasonable cost trade-off. It should be noted that in order to use a CMO, time would be required for technology transfer

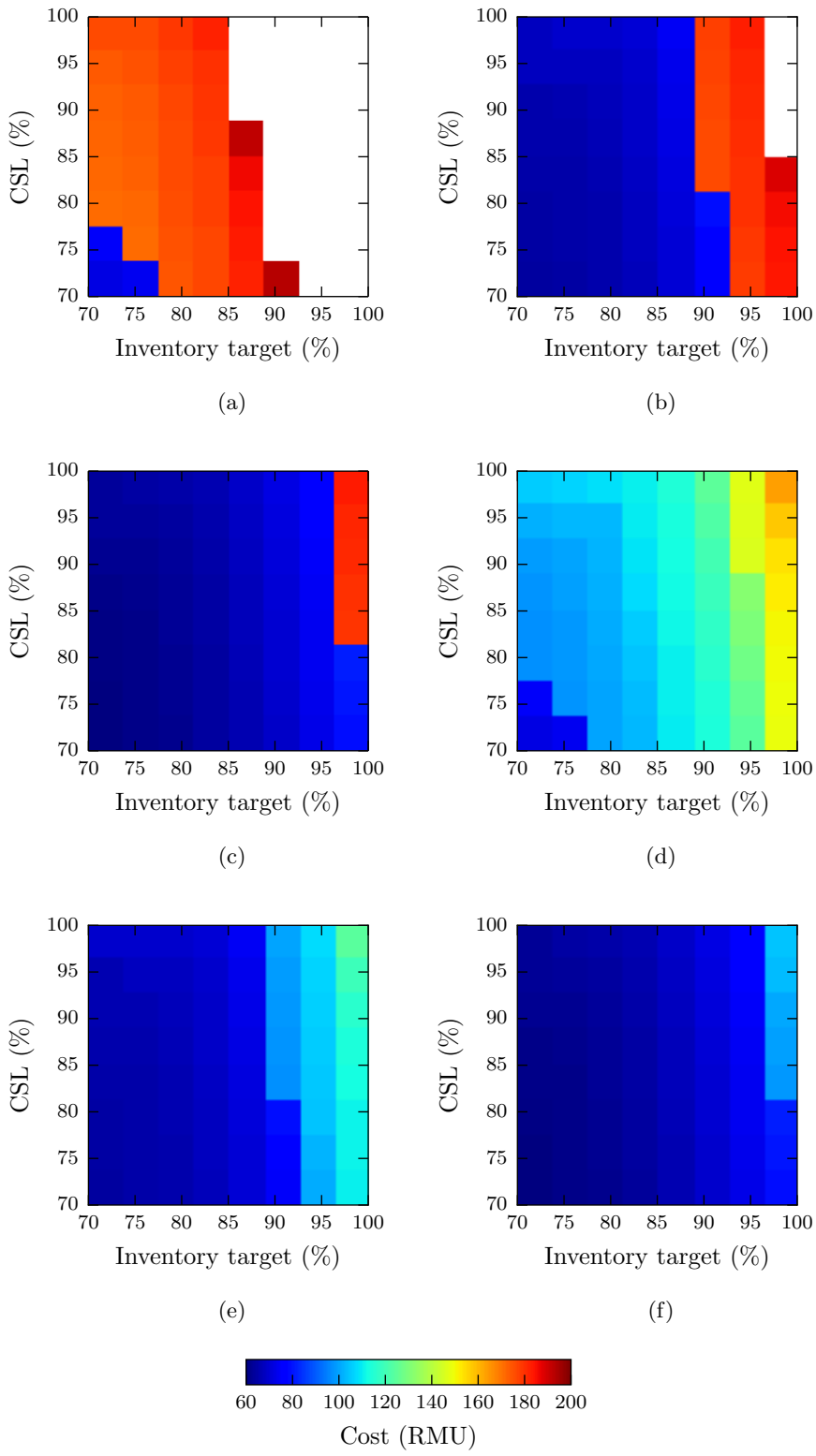


Figure 5.4: Heat maps illustrating the effect of titre fluctuations on antibody production in the first three years. Titres are (a) -20%, (b) base case, (c) +20%. This is repeated with CMO availability in (d)-(f).

and negotiating contracts. Hence, if a CMO is to be used, the decision needs to be made early on. A point to bear in mind is that these figures are showing the best options available over a three year capacity plan. If the planning horizon were to be extended to eight years, it may be that building new facilities is actually the best long-term option.

The multi-objective model can help decision makers determine the best operating points given their risk tolerance and long-term strategy. Variations in processes or operating procedures can influence the choices made. The results from this case study suggest that using a CMO is potentially the safest option for the first few years. There are reasons why a company may not want to use a CMO however, and thus the figures allow a decision maker to identify points where capacity bottlenecks could occur.

5.6 Summary

This chapter has explored the use of a multi-objective model in aiding a decision maker in choosing desired operating points for production planning. Both goal programming and the ϵ-constraint method were investigated as ways in which different solutions could be obtained. The ϵ-constraint method provided a more evenly distributed set of solutions, and allowed a user to more easily determine inflection points where capacity could be affected. Uncertainty surrounding biopharmaceutical production planning was explored using the multi-objective model to examine the robustness of solutions, allowing decision makers to identify the best schedules based on their risk tolerances.

This chapter has shown the merits of developing a tool which can quickly calculate optimal manufacturing schedules based on strategic criteria that a decision maker may enforce. It can also be easily adapted should one wish to consider other objectives or scenarios.

5.7 Nomenclature

Indices

g	goal (objective)
dev	goal deviation

Sets

T^{d}	time periods which have a demand
T^{inv}	time periods which have a stategic inventory level

Parameters

P_t^{inv}	number of products which have a strategic inventory level at time period t
P_t^{d}	number of products which have a demand at time period t
GT_g	goal target
$w_{g,\mathrm{dev}}$	weight assigned to goal deviation

Positive Variables

GLT_{gt}	goal level at time period t
$GD_{g,\mathrm{dev}}$	goal deviation

Free Variables

GL_g	goal level
$GD_{g,\mathrm{dev}}^{\mathrm{norm}}$	normalised goal deviation

Chapter 6

Conclusions and Future Work

6.1 Introduction

The purpose of this thesis was to develop a framework which can be used to determine optimal capacity plans for biopharmaceutical production, and allow an end-user to investigate how uncertain input parameters can affect the result. In order to accurately model a capacity plan, a biopharmaceutical strategist must consider the following:

- The process details including yield and duration for upstream and downstream production of each product.
- Manufacturing capabilities in each suite, and the network structure regarding transferral of intermediate product.
- Strategic options available to the company regarding capacity expansion, for example, outsourcing to a CMO or building new facilities.
- The multiple strategic criteria by which a capacity plan is optimised for.
- Financial requirements of manufacturing the therapeutics being modelled.
- Capital expenditure requirements for retrofitting or building facilities.
- Operational constraints between different products.

- The parameters which demonstrate uncertainty in the manufacturing and capacity planning stages.

This thesis has addressed all of these features of biopharmaceutical capacity planning in detail. Results are shown on how uncertainty surrounding CMO cost and product titres affect the capacity plans in question, and the overall costs over a finite period of time. The complexity of the model resulted in computational inefficiencies which were addressed by reformulating the model. Although the new formulation was a large improvement over the original model, there were still computational issues surrounding certain case studies. An extension to this work would investigate alternative formulations which would improve performance even further. The multi-criteria nature of biopharmaceutical manufacturing was addressed via the use of multi-objective optimisation. Three objectives were considered, with results showing the trade-offs surrounding the costs and risks associated with operating at various design points. The number of objectives could be increased further, but would compound the computational effort required to obtain sets of optimal solutions. The graphical representation of higher dimensional data would also be more difficult to understand. Lastly, the optimisation of biopharmaceutical capacity plans is an important aspect to biomanufacturing. As portfolios of products and the number of available manufacturing suites increase in size, the problem becomes non-trivial to solve. Therefore, computational tools which intelligently obtain optimal manufacturing schedules, as shown in this work, is of great relevance to industry.

6.2 Contributions of this thesis

The primary aim of this work is the creation of a mathematical model which obtains the optimal manufacturing schedules and capacity plans for a portfolio of products displaying different modes of cell-culture, amongst a network of multi-purpose facilities. Various constraints inherent in biopharmaceutical manufacturing make this computationally difficult. The work conducted to solve this problem and the contributions of this thesis are outlined below.

6.2.1 Capacity planning for batch and perfusion bioprocesses across multiple biopharmaceutical facilities

Chapter 3 describes the development of a novel model which includes both perfusion and fed-batch processes in its capacity planning capabilities. In order to correctly accommodate the differences between different process modes, sequence-dependent changeovers were introduced, adding further complexity to the model. The introduction of decoupled upstream and downstream production added further flexibility to the model, and more realistically represented biomanufacturing of perfusion based products.

A rolling time horizon procedure was used to obtain solutions using fewer computational resources, and was shown to provide capacity plans with lower overall costs. An industrial case study was analysed using the model to demonstrate the usefulness of a computational framework that optimises manufacturing schedules in a holistic manner that considers capital investment.

6.2.2 Biopharmaceutical Capacity Planning using a State Task Network Topology

Chapter 4 demonstrates the performance improvement obtained by reformulating the mathematical model into a state task network. Two case studies were used to show the merits of the reformulation in terms of model size and performance. In all cases, the state task network (STN) formulation was faster and provided better solutions.

The STN also added extra features to the model that increased the realism of the manufacturing schedules. Retrofitting downtime was included, and was shown to have a large impact on the solutions that were obtained. Increased flexibility in the way in which CMOs were modelled also changed the optimal solutions, resulting in cases where multiple CMOs were utilised during periods of high demand and restricted in-house capacity.

An investigation into the effect of changing the time resolution of the STN to two months was carried out, and was shown to drastically reduce the computa-

tional effort required. However, depending on the case study, it did not always provide better solutions overall.

Various strategic options surrounding how best to expand capacity were investigated. A decentralised expansion programme was compared to a centralised programme, and shown to be more robust when increased changeover durations were introduced. The issue of vastly different equipment in perfusion and fed-batch processes was examined in terms of retrofitting options. Executive decisions which prevent a suite from being multi-purpose were shown to be sub-optimal both in terms of the extra capital expenditure that would be required otherwise, and the increased risk involved in building new facilities to satisfy demands. Finally, the model was used to demonstrate its de-bottlenecking capabilities, and its usefulness in aiding a strategic planner in predicting how small changes in the manufacturing schedule affect long-term planning.

6.2.3 Multi-Criteria Strategic Planning for Biopharmaceutical Production

Chapter 5 continued with the development of the STN in Chapter 4 by adding a multi-objective component that could take into account the multi-criteria aspect of biomanufacturing. The ϵ-constraint method was compared to weighted-sum goal programming, and was shown to be the superior multi-objective method for generating complete sets of optimal solutions. It was then used to show how variations in certain input parameters could affect the design space and points at which capacity expansion was required.

6.3 Recommendations for Future Work

Creating a framework which closely represents the intricate details of biomanufacturing is a difficult task. Ways in which the framework developed here could be extended include:

- Problem features

- Uncertainty
- Alternative search heuristics

This section discusses how the framework could be extended to consider each of the above points.

6.3.1 Problem features

The mathematical model presented in this thesis is a close representation of real biomanufacturing for both perfusion and fed-batch processes. However, there are areas which could be extended in order to add further realism to the model. One of the issues with adding more functionality is that the computational effort will also be increased. Computational complexity is often a limiting factor in deciding which features to include, and thus new methods of improving the performance of the model should also be investigated. Some features which would enhance the model are discussed below.

Timing considerations

This model has used sequence-dependent changeover times to take into account the additional complexity of switching between fed-batch and perfusion processes. However, these times cannot exceed one time period, and thus if one wished to include extra setup times into the model (for example, extended seed train times), this would not be possible with the current formulation. Changeover times could be measured in terms of time periods instead of days, but that would not be a feasible strategy for long-term capacity planning where time periods are one month or greater. Alternatively, since idle time is considered a pseudo-product in the STN model, one could enforce downtime associated with changeovers by starting a campaign of the pseudo-product. This would also require changing the formulation of the model such that it allows a maximum of two products to be produced in a time period. A potential reformulation could be taken from examples of the proportional lot size and scheduling problem, where any unused time within a time period can be used for scheduling a second product. This

type of reformulation may also aid with increasing the flexibility of scheduling, especially when capacity is limited and multiple short campaigns of different products is required.

The number of time periods in the model has a large effect on computational efficiency. Efforts were made to reduce the number of time periods by making the time horizon two months rather than one. Whilst this did help obtain solutions faster, it led to inferior solutions when capacity was limited. A hybrid model could be useful in these situations. One model could use monthly time horizons, and another could use 2 month or quarterly horizons. Then the user can pick which model should be used in various parts of the 10 year capacity plan. A simple case would be to say that the finer model should be used at the beginning (perhaps the first year) when demands are more certain, and then use the less fine model in subsequent years. A hybrid model could also be used in a more systematic way. For example, use the fine model for the first year, and for the last 9 years use the coarse model. If there are any capacity bottlenecks in the subsequent 9 years, rerun the optimisation, but this time using the fine model in the year that had the bottleneck. This can be repeated until there are no bottlenecks, or until the entire 10 years is using the fine model. Given the performance improvement of using a coarser model, this method will not be as computationally expensive as it may first seem, and could provide better solutions faster in most cases.

Transportation duration was not included in the mathematical model, but would be an important factor when capacity is tight and intermediate inventory levels are low. Moving material between countries which are far apart would obviously require time and careful logistic planning. The model could be adapted to include transportation times by changing Equation 4.45 such that Q_{ijpt} becomes $Q_{ijp,t+\tau_{ij}^{\text{transport}}}$, where $\tau_{ij}^{\text{transport}}$ is the transportation duration between suites i and j.

Fixed costs

Currently, fixed costs are assigned to a particular suite a priori, and if that suite is used, the cost is applied in the objective function. This assumption is valid

if the fixed costs do not vary much between different products, but otherwise a different approach may be more suitable. Siganporia et al. (2012) use a model which calculates the fixed costs dynamically based on which products have been manufactured in a given suite. However, the performance hit from having extra constraints was deemed to outweigh the benefit of having this type of fixed cost model.

Objectives

This work introduced equations which considered objectives including total manufacturing cost, maintaining strategic inventory levels, and maximising customer service levels. The model could be extended by considering alternative criteria of interest to a strategic planner. Often, manufacturers prefer to have campaigns of longer duration rather than switching many times between products. Longer campaigns are easier to manage logistically and can minimise the risk of cross-contamination. Therefore, an objective which minimises the number of changeovers could be of interest. Alternatively, one could minimise the amount of downtime in a year attributed to changeovers, which is perhaps a more intuitive way of thinking about the same problem. Companies often have utilisation targets for their facilities, thus an objective could be introduced which minimises the distance between the targets and current utilisation levels. Capital expenditure is also high on the agenda for many biopharmaceutical companies, thus an objective which could minimise investment may also be of interest. In this work, capital investment is included in the total costs objective, thus moving this to a separate objective may be a more useful measure of performance, depending on a company's priorities.

6.3.2 Uncertainty

Uncertainty has been introduced into this work via the use of scenario based analysis, whereby input parameters are varied and the model is re-optimised. Whilst this is a useful technique, one may wish to obtain results which also have probabilities assigned to them. That is, given a probability distribution

of demands, one may want to know what the likelihood is of building a new facility. This is only possible if the input parameters are assigned probabilities. There are different ways in which uncertainty could be introduced. One method would be to randomly choose an input parameter's value based on a probability distribution, and then run the optimisation. This process is then repeated many times, such that a set of solutions is obtained, each with probability assigned to them. Another method would be to obtain a base case solution, and then measure its robustness by fixing the solution and varying the input parameters. The model would then be able to determine things such as how likely the customer service level is to be 100% given uncertain titres.

Another area where the model could be expanded is in considering attrition rates in clinical trials. The current model assumes that all products in its portfolio have or will have successfully passed their clinical trials. The development of biopharmaceutical drugs is inherently risky, with just 10-20% of drug candidates successfully reaching FDA approved status (Nie, 2015). Therefore, one could leverage this information to provide capacity plans where the probability of drugs reaching the market is also considered. In this way, the expected net present value could be obtained for optimal production plans. However, one of the reasons this idea was not built upon in this work is because of the added computational complexity required. The model was already hitting performance bottlenecks, so adding an extra layer would only deteriorate the situation. If further performance improvements are found, then including clinical trial attrition rates could be of interest to biopharmaceutical manufacturers.

6.3.3 Alternative search heuristics

Whilst mathematical programming provides a precise way of defining and solving complex problems, it can quickly become computationally intractable for larger problem sizes. Therefore, it may be useful to investigate other intelligent search techniques for determining optimal manufacturing schedules. Genetic algorithms are commonly used heuristics, but up till now have not been applied to biopharmaceutical capacity planning. An overview of a genetic algorithm that was

developed and used in this work is presented in Appendix A. Unfortunately, it did not provide better or faster solutions that the MILP model. Changing the chromosome structure or reformulating the way in which a genetic algorithm is used for capacity planning are two possible ways of improving the algorithm, and could be the basis of future work.

In summary, the work in this thesis provides a strong base for future work in the area of developing advanced capacity planning models for the sector.

References

Abara, J. (1989). Applying integer linear programming to the fleet assignment problem. *INTERFACES*, 19(4):20–28.

Acuna, J., Hewitt, M., Johnston, R., Kirkland, D., Shikibu, T., and Zhang, D. (2011). Modelling perfusion processes in biopharmaceutical production. *BioProcess International*, 9:52–59.

Adams, C. P. and Brantner, V. V. (2010). Spending on new drug development. *Health Economics*, 19(2):130–141.

Amanullah, A., Otero, J. M., Mikola, M., Hsu, A., Zhang, J., Aunins, J., Schreyer, H. B., Hope, J. A., and Russo, A. P. (2010). Novel micro-bioreactor high throughput technology for cell culture process development: Reproducibility and scalability assessment of fed-batch cho cultures. *Biotechnology and Bioengineering*, 106(1):57–67.

Amaro, A. and Barbosa-Póvoa, A. (2008). Planning and scheduling of industrial supply chains with reverse flows: A real pharmaceutical case study. *Computers & Chemical Engineering*, 32(11):2606 – 2625.

Amodeo, L., Chen, H., and El Hadji, A. (2007). Multi-objective supply chain optimization: An industrial case study. In Giacobini, M., editor, *Applications of Evolutionary Computing*, volume 4448 of *Lecture Notes in Computer Science*, pages 732–741. Springer Berlin Heidelberg.

Ashouri, P. (2011). *A Dynamic Simulation Framework for Biopharmaceutical Capacity Management*. PhD thesis, University College London.

Barry, F. (2015). Pfizer considers splitting company or going after M&A.

Bashiri, M., Khorasani, H., and Shiri, M. (2014). Multi objective supply chain network design considering customer satisfaction. In *Industrial Engineering and Engineering Management (IEEM), 2014 IEEE International Conference on*, pages 923–927.

Benzi, G. and Ceci, A. (1998). The 'drug value' in the european pharmaceutical system. *Pharmacological Research*, 37(5):333 – 337.

Berning, G., Brandenburg, M., Grsoy, K., Kussi, J. S., Mehta, V., and Tölle, F.-J. (2004). Integrating collaborative planning and supply chain optimization for the chemical process industry (i) - methodology. *Computers & Chemical Engineering*, pages 913–927.

Blackwell, J., Ransohoff, T. C., and Levine, H. L. (2010). Monoclonal antibody manufacturing strategies for the 21st century. *Pharmaceutical Outsourcing*, 11:28–35.

Blum, C. and Li, X. (2008). *Swarm Intelligence: Introduction and Application*, volume Swarm Intelligence in Optimization. Springer.

Cacciuttolo, M. (2007). Perfusion or fed-batch? A matter of perspective. *Cell culture and upstream processing*, page 173184.

Castro, P. M., Grossmann, I. E., and Novais, A. Q. (2006). Two new continuous-time models for the scheduling of multistage batch plants with sequence dependent changeovers. *Industrial & Engineering Chemistry Research*, 45(18):6210–6226.

Chinneck, J. W. (2000). Practical optimization: A gentle introduction.

Clincke, M.-F., Mölleryd, C., Zhang, Y., Lindskog, E., Walsh, K., and Chotteau, V. (2013). Very high density of cho cells in perfusion by ATF or TFF in WAVE bioreactor. Part I. Effect of the cell density on the process. *Biotechnology Progress*, 29:754767.

Colvin, M. and Maravelias, C. T. (2010). Modeling methods and a branch and cut algorithm for pharmaceutical clinical trial planning using stochastic programming. *European Journal of Operational Research*, 203(1):205 – 215.

Corporation, G. D. (2012). *CPLEX 12 Solver Manual*.

Dantzig, G. B. (1951). Maximization of a linear function of variables subject to linear inequalities. In Koopmans, T., editor, *Activity Analysis of Production and Allocation*. Wiley, New York.

Darby-Dowman, K. and Wilson, J. M. (2002). Developments in linear and integer programming. *The Journal of the Operational Research Society*, 53(10):pp. 1065–1071.

DiMasi, J. A. and Grabowski, H. G. (2007). The cost of biopharmaceutical R&D: is biotech different? *Managerial and Decision Economics*, 28(4-5):469–479.

Dorigo, M. (1992). *Optimization, Learning and Natural Algorithms*. PhD thesis, Dipartimento di Elettronica, Politecnico di Milano, Italy.

Dorigo, M. and Gambardella, L. M. (1997). Ant colony system: A cooperative learning approach to the traveling salesman problem. *IEEE Transactions on Evolutionary Computation*, 1:53–66.

Dorigo, M., Maniezzo, V., Colorni, A., Dorigo, M., Dorigo, M., Maniezzo, V., Maniezzo, V., Colorni, A., and Colorni, A. (1991). Positive feedback as a search strategy. Technical report.

Dréo, J. (2006). `http://en.wikipedia.org/wiki/File:Aco_branches.svg`.

Edwards, J. (2011). Patent cliff losses mean eli lilly must acquire or be acquired.

Eraslan, E. and Derya, T. (2010). Daily newspaper distribution planning with integer programming: an application in turkey. *Transportation Planning and Technology*, 33(5):423–433.

Ernst & Young (2015). Beyond borders - Biotechnology Industry Report 2015.

Farid, S. S. (2007). Process economics of industrial monoclonal antibody manufacture. *Journal of Chromatography B*, 848(1):8 – 18.

Foo, F., Karri, S., Davies, E., Titchener-Hooker, N., and Dunnill, P. (2001). Biopharmaceutical process development: Part i, information from the first product generation. *Pharmaceutical Technology Europe*, 13(6):58+60+62–64. cited By (since 1996) 8.

Gatica, G., Papageorgiou, L., and Shah, N. (2003a). Capacity planning under uncertainty for the pharmaceutical industry. *Chemical Engineering Research and Design*, 81(6):665 – 678.

Gatica, G., Shah, N., and Papageorgiou, L. G. (2003b). An aggregation approach for capacity planning under uncertainty for the pharmaceutical industry. *Foundations of Computer-Aided Process Operations*, 4:245–248.

George, E., Titchener-Hooker, N. J., and Farid, S. S. (2007). A multi-criteria decision-making framework for the selection of strategies for acquiring biopharmaceutical manufacturing capacity. *Computers & Chemical Engineering*, 31(8):889 – 901. 7th World Congress of Chemical Engineering.

George, E. D. and Farid, S. S. (2008a). Stochastic combinatorial optimization approach to biopharmaceutical portfolio management. *Industrial & Engineering Chemistry Research*, 47(22):8762–8774.

George, E. D. and Farid, S. S. (2008b). Strategic biopharmaceutical portfolio development: An analysis of constraint-induced implications. *Biotechnology Progress*, 24(3):698–713.

Goldberg, D. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional.

Guillén, G. and Grossmann, I. (2010). A global optimization strategy for the environmentally conscious design of chemical supply chains under uncertainty in the damage assessment model. *Computers & Chemical Engineering*, 34(1):42 – 58.

Guillén, G., Mele, F., Bagajewicz, M., Espuña, A., and Puigjaner, L. (2005). Multiobjective supply chain design under uncertainty. *Chemical Engineering Science*, 60(6):1535 – 1553.

Guntsch, M. and Middendorf, M. (2001). Pheromone modification strategies for ant algorithms applied to dynamic TSP. In *Proceedings of the EvoWorkshops on Applications of Evolutionary Computing*, pages 213–222, London, UK, UK. Springer-Verlag.

Guntsch, M. and Middendorf, M. (2003). Solving multi-criteria optimization problems with population-based ACO. In *Proceedings of the 2nd international conference on Evolutionary multi-criterion optimization*, EMO'03, pages 464–478, Berlin, Heidelberg. Springer-Verlag.

Habib, S. J. and Marimuthu, P. N. (2010). Optimized capacity planning and performance measurement through OPNET modeler. In *International Conference on Computer Applications and Industrial Electronics*.

Hwang, C. and Masud, A. (1979). *Multiple Objective Decision Making - Methods and Applications: A State of the Art Survey*, volume 164 of *Lecture Notes in Economics and Mathematical Systems*. Springer-Verlag.

Inman, P. and Hawkes, A. (2011). Pfizer to close factory with loss of 2,400 jobs as manufacturing soars.

Kantorovich, L. V. (1960). Mathematical methods of organizing and planning production. *Management Science*, 6(4):pp. 366–422.

Karmarkar, N. (1984). A new polynomial-time algorithm for linear programming. *Combinatorica*, 4:373–395. 10.1007/BF02579150.

Kelley, B. (2009). Industrialization of mab production technology: the bioprocessing industry at a crossroads. *mAbs*, 1(5):443–452.

Klee, V. and Minty, G. J. (1972). How good is the simplex algorithm? In Third Sympos. (Univ. California, CA, ., editor, *Inequalities, III*, pages 159–175, New York. Academic Press.

Kondili, E., Pantelides, C., and Sargent, R. (1993). A general algorithm for short-term scheduling of batch operations - I. MILP formulation. *Computers & Chemical Engineering*, 17(2):211 – 227. An International Journal of Computer Applications in Chemical Enginnering.

Ku, H. M. and Karimi, I. A. (1991). Scheduling algorithms for serial multiproduct batch processes with tardiness penalties. *Computers & Chemical Engineering*, 15(5):283 – 286.

Kux, L. (2012). Prescription drug user fee rates for fiscal year 2013. *Federal Register*, 77:45639–45643.

Lakhdar, K. and Papageorgiou, L. G. (2008). An iterative mixed integer optimisation approach for medium term planning of biopharmaceutical manufacture under uncertainty. *Chemical Engineering Research and Design*, 86(3):259 – 267.

Lakhdar, K., Savery, J., Papageorgiou, L., and Farid, S. S. (2007). Multiobjective long-term planning of biopharmaceutical manufacturing facilities. *Biotechnology Progress*, 23(6):1383–93.

Lakhdar, K., Zhou, Y., Savery, J., Titchener-Hooker, N. J., and Papageorgiou, L. G. (2005). Medium term planning of biopharmaceutical manufacture using mathematical programming. *Biotechnology Progress*, 21(5):1478–1489.

Lakshmikanthan, J. (2007). Outsourcing: Biologics manufacturing: The CMO advantage. *BioPharm International*, 20.

Langer, E. S. (2011). Survey examines trends in the outsourcing relationship. *BioPharm International*, 24:26–33.

Lázaro, M., na, A. E., and Puigjaner, L. (1989). A comprehensive approach to production planning in multipurpose batch plants. *Computers & Chemical Engineering*, 13(9):1031 – 1047.

Lipsky, M. S. and Sharp, L. K. (2001). From idea to market: the drug approval process. *American Board of Family Practice*, 14:362–367.

Lorigeon, T., Billaut, J.-C., and Bouquard, J.-L. (2002). A dynamic programming algorithm for scheduling jobs in a two-machine open shop with an availability constraint. *The Journal of the Operational Research Society*, 53(11):pp. 1239–1246.

Lundy, M. and Mees, A. (1986). Convergence of an annealing algorithm. *Mathematical Programming*, 34:111–124. 10.1007/BF01582166.

Maravelias, C. T. and Grossmann, I. E. (2001). Simultaneous planning for new product development and batch manufacturing facilities. *Industrial & Engineering Chemistry Research*, 40(26):6147–6164.

Marichal-Gallardo, P. A. and Álvarez, M. M. (2012). State-of-the-art in downstream processing of monoclonal antibodies: Process trends in design and validation. *Biotechnology Progress*, 28(4):899–916.

Mavrotas, G. (2009). Effective implementation of the epsilon-constraint method in multi-objective mathematical programming problems. *Applied Mathematics and Computation*, 213(2):455–465.

Mavrotas, G. and Florios, K. (2013). An improved version of the augmented ϵ-constraint method (augmecon2) for finding the exact pareto set in multi-objective integer programming problems. *Applied Mathematics and Computation*, 219(18):9652 – 9669.

Merkle, D., Middendorf, M., and Schmeck, H. (2000). Ant colony optimization for resource-constrained project scheduling. In *IEEE Transactions on Evolutionary Computation*, pages 893–900. Morgan Kaufmann.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of State Calculations by Fast Computing Machines. *J. Chem. Phys.*, 21:1087–1092.

Meuwly, F., Weber, U., Ziegler, T., Gervais, A., Mastrangeli, R., Crisci, C., Rossi, M., Bernard, A., von Stockar, U., and Kadouri, A. (2006). Conversion

of a CHO cell culture process from perfusion to fed-batch technology without altering product quality. *Journal of Biotechnology*, 123(1):106 – 116.

Meyer, B. and Ernst, A. (2004). Integrating ACO and constraint propagation. In Dorigo, M., Birattari, M., Blum, C., Luca, Mondada, F., and Stützle, T., editors, *Ant Colony, Optimization and Swarm Intelligence: 4th International Workshop, ANTS 2004*, volume 3172 of *Lecture Notes in Computer Science*, pages 166–177. Springer Verlag GmbH.

Monma, C. L. and Potts, C. N. (1989). On the complexity of scheduling with batch setup times. *Operations Research*, 37:798–804.

Moreno, M. S. and Montagna, J. M. (2009). A multiperiod model for production planning and design in a multiproduct batch environment. *Mathematical and Computer Modelling*, 49(7-8):1372 – 1385.

Nie, W. (2015). *Cost Evaluation and Portfolio Management Optimization for Biopharmaceutical Product Development*. PhD thesis, UCL.

Osman, M., Abo-Sinna, M., Amer, A., and Emam, O. (2004). A multi-level non-linear multi-objective decision-making under fuzziness. *Applied Mathematics and Computation*, 153(1):239 – 252.

Papageorgiou, L. G. and Pantelides, C. C. (1996). Optimal campaign planning/scheduling of multipurpose batch/semicontinuous plants. 1. Mathematical formulation. *Industrial & Engineering Chemistry Research*, 35(2):488–509.

Papageorgiou, L. G., Rotstein, G. E., and Shah, N. (2001). Strategic supply chain optimization for the pharmaceutical industries. *Industrial & Engineering Chemistry Research*, 40(1):275–286.

Paparrizos, K., Samaras, N., and Stephanides, G. (2003). A new efficient primal dual simplex algorithm. *Computers & Operations Research*, 30(9):1383 – 1399.

Paul, S. M., Mytelka, D. S., Dunwiddie, C. T., Persinger, C. C., Munos, B. H., Lindborg, S. R., and Schacht, A. L. (2010). How to improve R&D productivity:

the pharmaceutical industry's grand challenge. *Nat Rev Drug Discov*, 9(3):203–214.

Pishvaee, M. S. and Razmi, J. (2012). Environmental supply chain network design using multi-objective fuzzy mathematical programming. *Applied Mathematical Modelling*, 36(8):3433 – 3446.

Pohlscheidt, M., Jacobs, M., Wolf, S., Thiele, J., Jockwer, A., Gabelsberger, J., Jenzsch, M., Tebbe, H., and Burg, J. (2013). Optimizing capacity utilization by large scale 3000 L perfusion in seed train bioreactors. *Biotechnology Progress*, 29:222–229.

Pollock, J., Ho, S. V., and Farid, S. S. (2013). Fed-batch and perfusion culture processes: Economic, environmental, and operational feasibility under uncertainty. *Biotechnology and Bioengineering*, 110(1):206–219.

Pozo, C., Ruz-Femenia, R., Caballero, J., Guilln-Goslbez, G., and Jimnez, L. (2012). On the use of principal component analysis for reducing the number of environmental objectives in multi-objective optimization: Application to the design of chemical supply chains. *Chemical Engineering Science*, 69(1):146 – 158.

Ramteke, M. and Srinivasan, R. (2011). Integrating graph-based representation and genetic algorithm for large-scale optimization: Refinery crude oil scheduling. *Computer Aided Process Engineering*, 29:567–571.

Ransohoff, T. C. (2004). Considerations impacting the make vs. buy decision. *American Pharmaceutical Outsourcing*.

Roghanian, E., Sadjadi, S., and Aryanezhad, M. (2007). A probabilistic bi-level linear multi-objective programming problem to supply chain planning. *Applied Mathematics and Computation*, 188(1):786 – 800.

Rosenthal, R. E. (2011). *GAMS - A User's Guide*. GAMS Development Corporation, Washington, DC, USA.

Rotstein, G., Papageorgiou, L., Shah, N., Murphy, D., and Mustafa, R. (1999). A product portfolio approach in the pharmaceutical industry. *Computers & Chemical Engineering*, 23(Supplement 1):S883 – S886. European Symposium on Computer Aided Process Engineering, Proceedings of the European Symposium.

Shmygelska, A., Hernández, R. A., and Hoos, H. H. (2002). An ant colony optimization algorithm for the 2d hp protein folding problem. In *Proceedings of the Third International Workshop on Ant Algorithms*, ANTS '02, pages 40–53, London, UK. Springer-Verlag.

Siganporia, C., Ghosh, S., Daszkowski, T., Papageorgiou, L. G., and Farid, S. S. (2012). Production planning of batch and semi-continuous bioprocesses across multiple biopharmaceutical facilities. In Bogle, I. D. L. and Fairweather, M., editors, *22nd European Symposium on Computer Aided Process Engineering*, volume 30 of *Computer Aided Chemical Engineering*, pages 377 – 381. Elsevier.

Siganporia, C. C., Ghosh, S., Daszkowski, T., Papageorgiou, L. G., and Farid, S. S. (2014). Capacity planning for batch and perfusion bioprocesses across multiple biopharmaceutical facilities. *Biotechnology Progress*, 30(3):594–606.

Sousa, R., Shah, N., and Papageorgiou, L. G. (2008). Supply chain design and multilevel planning–an industrial case. *Computers & Chemical Engineering*, 32(11):2643 – 2663. Enterprise-Wide Optimization.

Stonier, A., Pain, D., Westlake, A., Hutchinson, N., Thornhill, N. F., and Farid, S. S. (2013). Integration of stochastic simulation with multivariate analysis: Short-term facility fit prediction. *Biotechnology Progress*, 29(2):368–377.

Stützle, T. and Hoos, H. H. (2000). Maxmin ant system. *Future Generation Computer Systems*, 16:889–914.

Sun, Y. and Turnquist, M. A. (2007). Investment in transportation network capacity under uncertainty: Simulated annealing approach. *Transportation Research Record*, 2039:67–74.

Sung, C. and Maravelias, C. T. (2006). An attainable region approach for effective production planning. In Marquardt, W. and Pantelides, C., editors, *16th European Symposium on Computer Aided Process Engineering and 9th International Symposium on Process Systems Engineering*, volume 21 of *Computer Aided Chemical Engineering*, pages 1893 – 1898. Elsevier.

Tamiz, M., Mirrazavi, S., and Jones, D. (1999). Extensions of pareto efficiency analysis to integer goal programming. *Omega*, 27(2):179 – 188.

Tandon, M., Cummings, P., and Van, M. L. (1995). Scheduling of multiple products on parallel units with tardiness penalties using simulated annealing. *Computers & Chemical Engineering*, 19(10):1069 – 1076.

Tsenov, A. (2006). Simulated annealing and genetic algorithm in telecommunications network planning. *International Journal of Information and Mathematical Sciences*, 2:240–245.

Ündey, C., Ertunç, S., Mistretta, T., and Looze, B. (2010). Applied advanced process analytics in biopharmaceutical manufacturing: Challenges and prospects in real-time monitoring and control. *Journal of Process Control*, 20(9):1009 – 1018. ADCHEM 2009 Special Issue.

Urselmann, M., Sand, G., and Engell, S. (2009). A memetic algorithm for global optimization in chemical process synthesis. In *Proceedings of the Eleventh conference on Congress on Evolutionary Computation*, CEC'09, pages 1721–1728, Piscataway, NJ, USA. IEEE Press.

Vahdani, B., Tavakkoli-Moghaddam, R., Modarres, M., and Baboli, A. (2012). Reliable design of a forward/reverse logistics network under uncertainty: A robust-m/m/c queuing model. *Transportation Research Part E: Logistics and Transportation Review*, 48(6):1152 – 1168.

Varma, V. A., Pekny, J. F., Blau, G. E., and Reklaitis, G. V. (2008). A framework for addressing stochastic and combinatorial aspects of scheduling and

resource allocation in pharmaceutical R&D pipelines. *Computers & Chemical Engineering*, 32(45):1000 – 1015.

Walsh, G. (2014). Biopharmaceutical benchmarks 2014. *Nat Biotech*, 32(10):992–1000.

Wang, K.-J. and Chen, M.-J. (2009). Cooperative capacity planning and resource allocation by mutual outsourcing using ant algorithm in a decentralized supply chain. *Expert Systems with Applications*, 36(2, Part 2):2831 – 2842.

Zadeh, N. (2008). What is the worst case behavior of the simplex algorithm?

Appendix A

Genetic algorithm optimisation procedure

This section describes an attempt to determine whether a framework using genetic algorithms (GAs) could help improve performance over MILP formulations.

The first and arguably most important component of a genetic algorithm is how the problem is represented in the chromosome. Figure A.1 shows the chromosome structure that was chosen for this problem. Level one consists of an array of a fixed size. The size is dependent on the number of upstream and downstream suites. Each element in the array is yet another array (represented by level two), containing the allocation and duration of production. However, since multiple products can be manufactured in a suite over the time horizon being modelled, the array in level two must be variable in size. Thus, for a problem with two USP suites and two DSP suites, there would be one fixed array of four elements, each element being a variable sized array containing the product allocation and duration. In the example, p_1 is produced for 30 days, p_2 for 150 days, and so on. To allow for idle time, a pseudo-product was introduced (p_0). The total time allocated in each variable size array in level two must equal the time horizon being planned for.

There are various components to a genetic algorithm, but the three most fundamental include initialisation, evaluation, and mutation of the solutions. Each

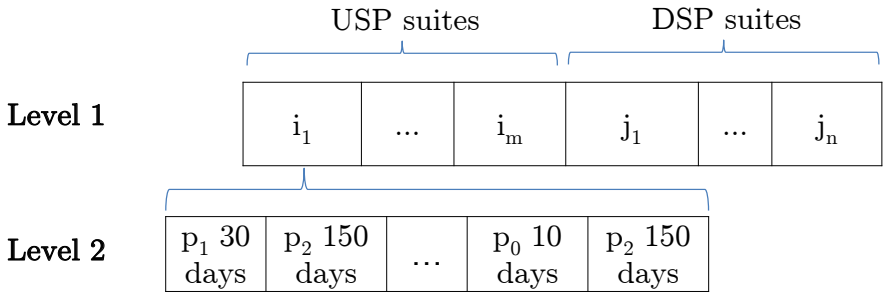


Figure A.1: Chromosome structure with an example of product allocation

will be discussed next.

Initialisation

The initialisation strategy can have a great influence on the optimisation procedure, especially the speed at which an optimal solution is found. However, depending on the strategy, it can also be too restrictive, leading to local optima being found rather than the globally optimal solution. Therefore, care must be taken to ensure that the genetic algorithm does not have a bias towards suboptimal solutions.

Here, a simple initialisation strategy is described, which is used to populate 25 initial solutions. Figure A.2 outlines the basis of the strategy. For each suite, the amount of time available is calculated. A product is chosen based on the amount of time available, and then the number of campaigns is chosen randomly. From this, the production duration is calculated, and then this allocation is added to the chromosome (in the variable array for this particular suite). This process will continue whilst there is still time available. Once no more time is left, the initialisation will move onto the next suite and repeat the process. The initialisation is over once all suites have been considered.

Evaluation

The purpose of the evaluation function is to calculate the total costs of a solution. In order to accomplish this, the solution must first be analysed to ensure that there is correct linking between upstream and downstream processes. This adds a lots of complexity to the algorithm, and would not be necessary if upstream

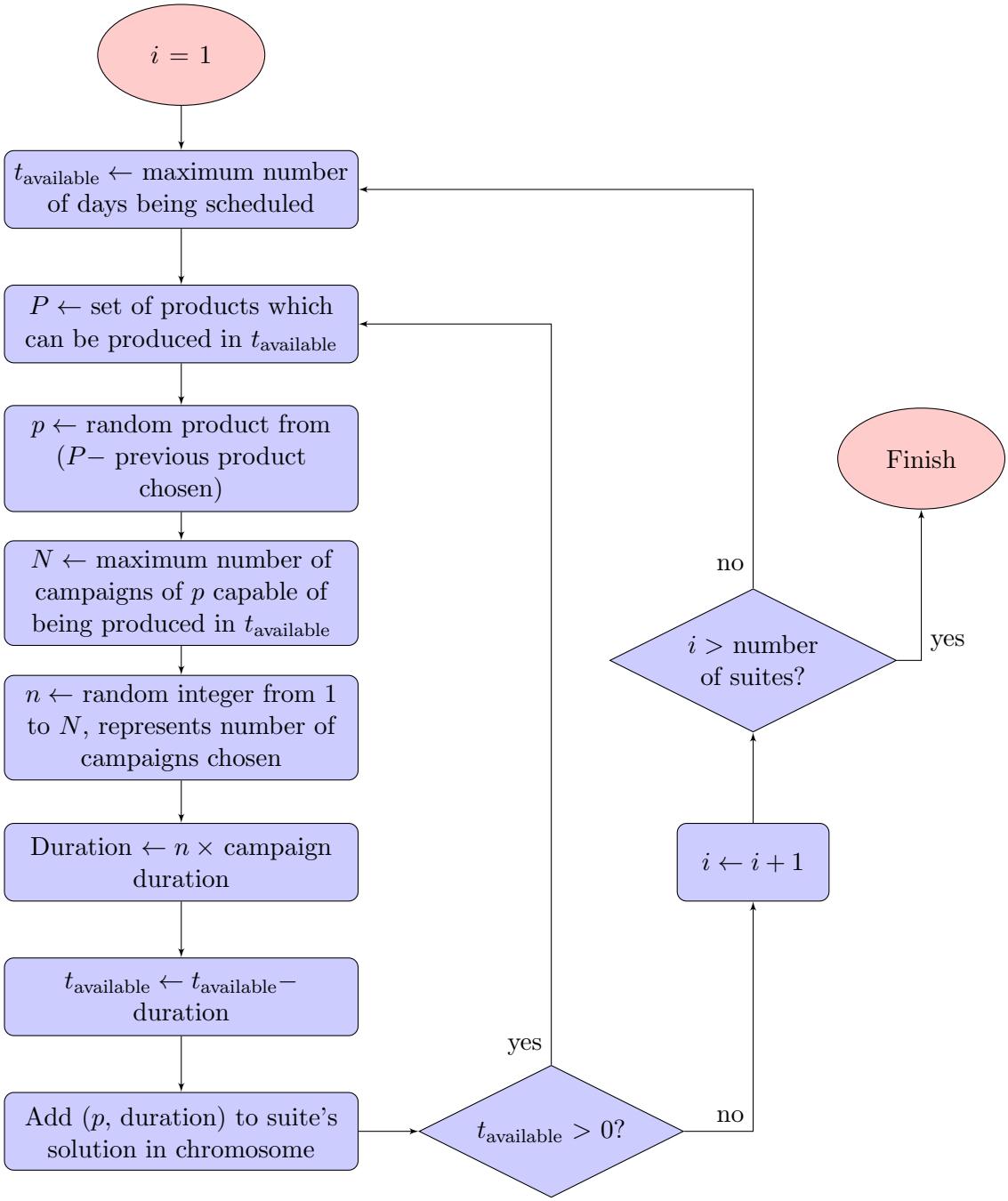


Figure A.2: Initialisation strategy

and downstream processes were modelled as one black-box process. These checks must be carried out either in the evaluation function, or in both the initialisation function and mutation operator.

Algorithm 1 shows the part of the evaluation function which deals with ensuring enough material is available in USP suites before being purified for downstream production. It first goes through each USP suite and calculates inventory levels for each product. Then, for every batch that is allocated in a DSP suite, it first checks that there is available USP material, and then adjusts USP inventory levels accordingly. Once all DSP batches have been accounted for, the USP inventory costs can be calculated. Demands are met via a similar process but for DSP inventory levels. Finally, the total cost is updated with the variable, inventory and backlog costs.

Since both the intermediate product and final product have shelf-lives, the material taken from the USP or DSP suites is always taken from the earliest possible time. Although not shown in Algorithm 1, when material from a USP suite is required, a transfer matrix for that particular DSP suite is used to determine the order that the USP suites should be analysed. This matrix is created when the framework begins, and uses lexicographic analysis to determine for every DSP suite which order of USP suites (based on transportation cost and time) material should be sourced from.

Mutation operator

The mutation operator is shown in Algorithm 2, and is simply a modification of the initialisation strategy. Instead of $t_{\text{available}}$ being the total available time for the planning horizon, here it is equal to the time that was allocated to the mutated product. In addition, the previous product is not excluded from the set of products that can be chosen. In this way, production durations of existing allocations can be extended. Finally, the solution is reduced such that allocations of the same product are combined into one allocation, thereby reducing the size of the variable array.

Algorithm 1 Extract of evaluation strategy

```
for each USP suite i do
  for each product allocated to i do
    Calculate when the batches are produced (considering ramp-up times)
    Calculate variable cost
    Calculate inventory levels (daily)
  end for
end for

for each DSP suite j do
  for each product allocated to j do
    Calculate when the batches are produced
    requiredAmount ← batchLoad
    day ← max(current day − USP shelf life, 1)
    while day ≤ current day and requiredAmount > 0 do
      for each USP suite in transfer matrix do
        x ← Take as much material as possible (up to requiredAmount)
        requiredAmount ← requiredAmount − x
        Adjust USP suite's inventory for this and every subsequent day
        if requiredAmount = 0 then
          break
        end if
      end for
      if requiredAmount > 0 then
        No DSP batches are produced at this time
        Reinstate any inventory that was subtracted from USP suites
      end if
      day ← day + 1
    end while
    Calculate inventory levels (daily)
    Calculate variable cost
  end for
end for

for each USP suite i do
  Calculate inventory cost for each product
end for

for each product do
  for each demand do
    requiredAmount ← demand
    day ← max(demand due date − DSP shelf life, 1)
    while day ≤ current day and requiredAmount > 0 do
      for each DSP suite do
        x ← Take as much material as possible (up to requiredAmount)
        requiredAmount ← requiredAmount − x
        Adjust DSP suite's inventory for this and every subsequent day
        if requiredAmount = 0 then
          break
        end if
      end for
      day ← day + 1
    end while
  end for
  demandPenalty ← requiredAmount × penalty
end for

for each DSP suite j do
  Calculate inventory cost for each product
end for

Cost ← variable costs + inventory costs + demand penalty costs
```

Algorithm 2 Mutation strategy

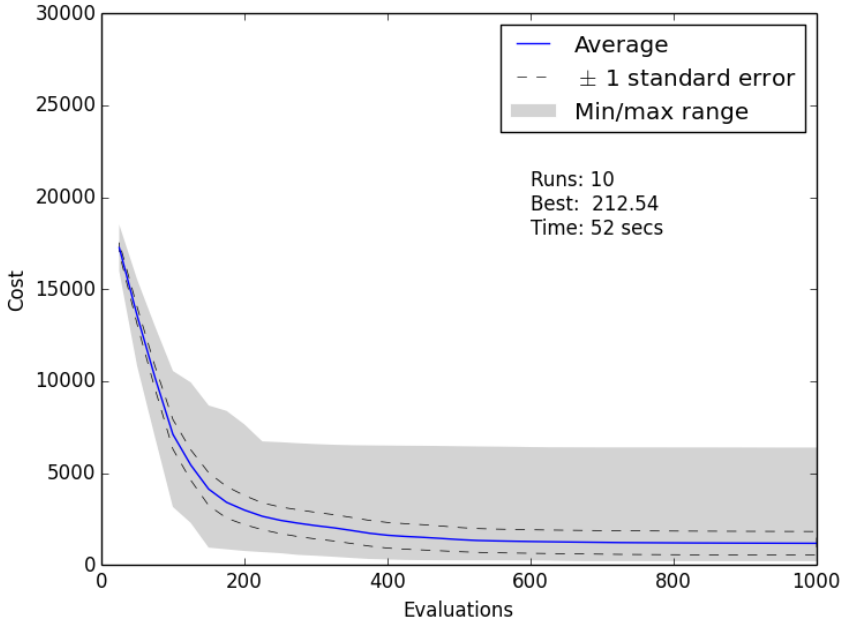
for each product allocation in chromosome **do**
  if mutate {based on probability, i.e. 20% chance of mutation} **then**
    Do as per initialisation strategy, except:
      $t_{\text{available}}$ = time allocated to current product
      P does not exclude previous product
    Reduce new products {p_1 150 days followed by p_1 150 days = p_1 300 days}
  end if
end for

Results

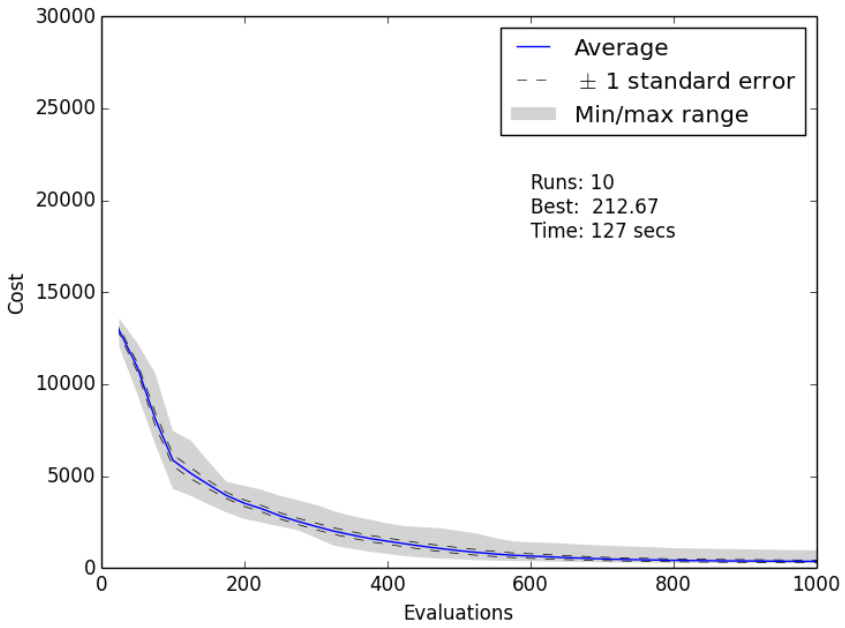
The computational results of a one year capacity plan are shown in Figure A.3. The number of evaluations run was 1000, and the total number of runs was 10. The average fitness (which in this case is the total cost) across all 10 runs at each evaluation is shown by a blue line. The range of values across the 10 runs is shown by the grey area. The best solution obtained and the total time required to run 1000 evaluations for 10 runs are also shown under the legend.

Two initialisation strategies were investigated, the first used the strategy shown in Figure A.2, whereas the second initialisation strategy applied checks on USP material prior to allocating DSP batches. That is, where as before the checks were made in the evaluation function, in the second strategy the checks were made in the initialisation and mutation functions. This led to a much narrower range, resulting in a lower average fitness value. However, the increased complexity of checking for USP material at each stage in the mutation operation led to an increase in computational time, and there was no improvement in the best fitness value obtained. For comparison, the one year capacity plan was also optimised via the STN mathematical model described in Chapter 4. The optimal solution was found to be 212.53, and this was obtained within 0.5 seconds.

Figure A.4 shows the comparison between the two initialisation/mutation strategies on two year capacity plans. Again, the range is tighter for the strategy which checks USP material inside the initialisation and mutation functions. However, the computational time has increased by more than four times, and the best solution found was much inferior to the simple strategy. So the results suggest that have a tighter range is not necessarily an important factor here, and does



(a)



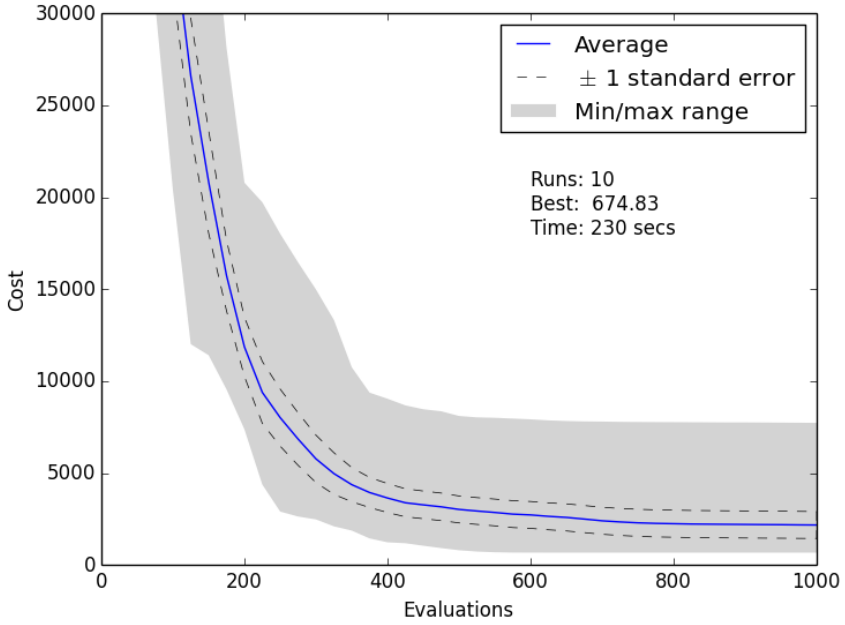
(b)

Figure A.3: Performance of GA for one year capacity plan. Optimisation profiles are shown for (a) standard initialisation/mutation and (b) checks on USP material during initialisation and mutation

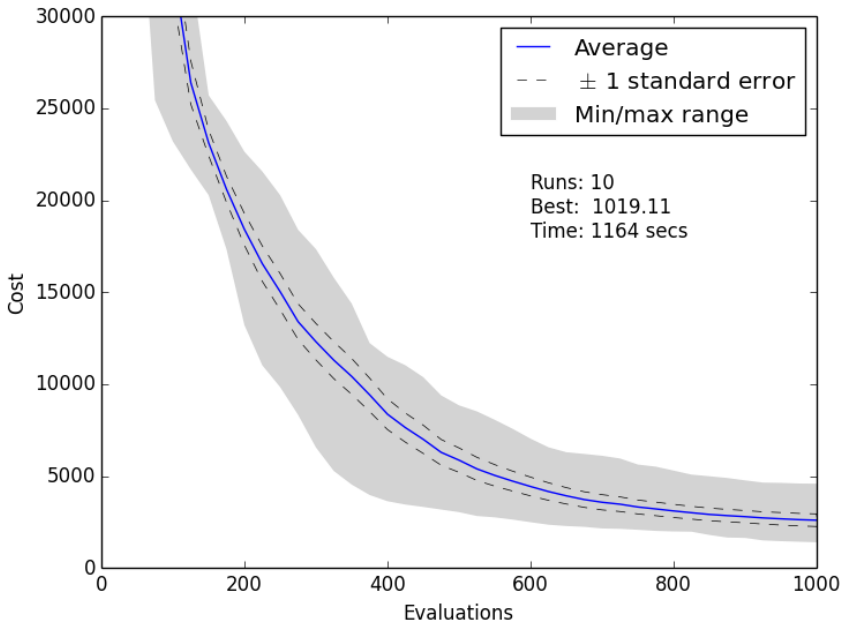
not aid in performance or solution quality as may initially be presumed. Again, the two year capacity plan was also optimised using a mathematical model. The best solution found was 671.91, and was obtained in 5 seconds.

Both the one and two year capacity plans were much quicker to solve using mathematical techniques rather than the genetic algorithm implementation shown here. Many improvements could be made to the initialisation and mutation algorithms, but the main issue is that the way the chromosome is structured restricts scalability. As the problem becomes larger and more years are planned for, the size of the variable length arrays become ever greater, and any loops within the mutation operation have a much larger impact on performance. Heuristics and tweaks to the mutation and initialisation algorithms may also lead to local optima, as was the case in Figure A.4b, where the best solution was not as good as the simpler initialisation/mutation strategy.

In theory, the GA model described here should be able to obtain better solutions than a discrete MILP model, simply because using a daily resolution allows for more manufacturing flexibility. Owing to performance issues, however, longer capacity plans (where this may be more noticeable) were not obtainable. Further work must be conducted to investigate whether using a genetic algorithm for this type of problem is of benefit, both in terms of solution quality and performance. The results obtained so far suggest that mathematical techniques are a better approach. It seems that if genetic algorithms are to be used, a different chromosome structure may be required, so that performance is not a limiting factor.



(a)



(b)

Figure A.4: Performance of GA for two year capacity plan. Optimisation profiles are shown for (a) standard initialisation/mutation and (b) checks on USP material during initialisation and mutation

Appendix B

Papers by the author

The following are papers that have been published by the author:

Siganporia, C., Ghosh, S., Daszkowski, T., Papageorgiou, L. G., and Farid, S. S. (2012). Production planning of batch and semi-continuous bioprocesses across multiple biopharmaceutical facilities. In Bogle, I. D. L. and Fairweather, M., editors, *22nd European Symposium on Computer Aided Process Engineering*, volume 30 of *Computer Aided Chemical Engineering*, pages 377-381. Elsevier.

Siganporia, C. C., Ghosh, S., Daszkowski, T., Papageorgiou, L. G., and Farid, S. S. (2014). Capacity planning for batch and perfusion bioprocesses across multiple biopharmaceutical facilities. *Biotechnology Progress*, 30(3):594-606.