SYNTHESIS OF BIOPROCESSES USING PHYSICAL PROPERTIES DATA

M.A. Steffens, E.S. Fraga and I.D.L. Bogle [1]
Department of Chemical Engineering, University College London.

Short title: Bioprocess Synthesis.

November 24, 2000

[1] Corresponding author: I.D.L. Bogle, Department of Chemical Engineering, University College London, Torrington Place, London, WC1E 7JE. Phone: +44 (0)171 419 3803. Fax: +44 (0)171 383 2348. Email: d.bogle@ucl.ac.uk.

Abstract

The aim of this paper is to illustrate and evaluate a synthesis technique for bioprocesses. Physical property information is used to screen candidate units thereby reducing the size of the synthesis problem. In this way only units which exploit large property differences between components in a stream are selected. This is important for bioprocesses because of the large number of components and wide range of unit operations which are available. The screening technique and bioprocess unit design methodologies have been incorporated within an implicit enumeration algorithm which was developed for chemical process synthesis and is implemented in the Java language. An important advantage is its ability to generate a ranked list of N flowsheets which may subsequently be analysed in more detail.

Two case studies are used to evaluate the bioprocess synthesis technique. The first system involves a product which is secreted from the host organism while the second has significantly different characteristics in that the product is intracellular and forms inclusion bodies. The second case study, in particular, is a large synthesis problem with 12 unit operations and 20 contaminant compounds considered. The results show that the synthesis methodology identifies a set of economically optimal flowsheets in a reasonable computational time which demonstrates its ability to deal with large synthesis problems. Using the synthesis methodology we can generate bioprocesses which are optimal in a system wide, rather than unit by unit, sense.

Keywords: process synthesis, bioprocesses, physical properties, optimisation.

1 Introduction

New product development is an integral part of the biochemical production industry due to the large range of biological products which are potentially valuable to society. Once a new product has been developed in the laboratory, however, a process which can produce and purify the product in large quantities is required. The success of a biochemical compounds manufacturing company may depend on the ability to design such a process in an economically optimal way (Leser and Asenjo, 1992; Wheelwright, 1987). For bioprocesses in particular, it is important to choose the optimal system as early as possible in the design procedure. This is because a process' characteristics are fixed once approved by the relevant regulatory body. Subsequent changes must go through a further expensive approval process (Petrides *et al.*, 1995).

Bioprocess flowsheets are often synthesised in a sequential fashion, proceeding from one unit to the next until product specifications are met (Wheelwright, 1987). Individual units are subsequently optimised to improve plant performance. Although this approach may produce economically adequate processes, alternative designs may be more profitable. To avoid this problem we require a systematic synthesis and design procedure which considers the overall process, rather than individual units (Wheelwright, 1987).

There currently exists a large body of literature on systematic process synthesis and design methods for chemical processes. These can be broadly classified into heuristic based approaches, such as the work by Douglas (1988), and algorithmic techniques (e.g. Fraga and McKinnon, 1994; Friedler *et al.*, 1996; Grossman and Kravanja 1995; Smith and Pantelides, 1995). The same abundance of literature does not exist for bioprocess synthesis techniques. Notable exceptions include work presented by Petrides (1994), who developed a synthesis procedure which uses expert knowledge to select unit operations. Lienqueo *et al.* (1996) describe a synthesis methodology which uses a combination of expert knowledge and physical properties data to synthesise downstream purification flowsheets. Both of these approaches are capable of synthesising economically favorable processes and represent significant advances in the field of bioprocess synthesis. However, two important characteristics of the synthesis techniques should be noted:

1. the use of expert knowledge may lead to the elimination of novel flowsheets which seem to contravene prevailing experience yet have interesting or desirable features.
2. both synthesis techniques treat the recovery and high resolution purification sections of a bioprocess as separate entities. This means that although each section of the plant may be optimally designed, the same cannot be said of the whole process.

Algorithms which are based on mathematical optimisation techniques provide a more rigorous approach to synthesis which allows us to avoid the above two shortcomings. However, when attempting to apply optimisation methods to solve the bioprocess synthesis problem we encounter several major obstacles:

- Unit operations commonly used for downstream purification generally separate components non-sharply. A non-sharp separation is one in which the feed components are distributed between the effluent streams.

- A large selection of unit operations is available.
- Biological streams generally contain a large number of compounds.

These characteristics lead to a large number of alternative feasible flowsheets and, therefore, a correspondingly large search space for the synthesis algorithm (Fraga, 1996b; Lienqueo *et al.*, 1996). Synthesis problems of this size are difficult to solve using numerical optimisation techniques.

Fraga (1998) described the *Jacaranda* synthesis package which is capable of tackling problems with the characteristics described above. It is based on the use of discretisation to convert the mixed integer nonlinear programming (MINLP) problem into a graph generation and search problem. Continuous variables, such as stream compositions and properties and unit design parameters, are mapped to a discrete space using discretisation parameters chosen by the user. In this paper we extend the Jacaranda process synthesis package to incorporate bioprocess synthesis.

On application of the technique, however, the authors found that the use of discretisation alone was not sufficient to make bioprocess synthesis problems tractable. Therefore, the number of synthesis alternatives is further reduced by screening candidate units using physical properties data. In this way we select units which best exploit the physical properties differences between the components in a bioprocess stream. The technique is evaluated by applying it to the synthesis of flowsheets for the purification of two industrially produced biochemical products.

2 The Bioprocess Synthesis Methodology

The Jacaranda system (Fraga, 1998) is a software implementation of an implicit enumeration procedure for automated synthesis. Based on a discrete programming approach, Jacaranda is suitable for solving the downstream purification flowsheet synthesis problem because it has the following features:

- Many numerical synthesis algorithms require the user to specify a superstructure containing all possible flowsheet alternatives beforehand (e.g. Grossman and Kravanja, 1995; Smith and Pantelides, 1995). Defining a superstructure is difficult for biochemical systems which have many components, a large number of possible units and use non-sharp separators. In addition, the optimal flowsheet is not guaranteed to be included within the superstructure. Jacaranda, however, simultaneously generates and searches the superstructure using a discrete implicit enumeration method.
- The screening methodology is easily incorporated into the search procedure and significantly reduces ($\approx 50\%$) the size of the superstructure as we will show later.
- Complex models may be included because a discrete optimisation is used thereby making the algorithm computationally efficient. This is particularly useful for bioprocesses which are often operated in a batch fashion (Petrides, 1994). Unit models for batch processes often consist of sets of time dependent differential equations which must

be integrated (e.g. Steffens *et al.*, 1999). In addition the search procedure imposes no requirements on the models convexity and continuity.

Jacaranda has been implemented in the JAVA language and uses two basic data structures, or classes, which are required to synthesise a process flowsheet: the Stream and UnitModel classes. As the name suggests the Stream class is a data structure which represents a process stream. For instance, in the synthesis of bioprocesses a stream class may contain information on the concentrations of each component (e.g. cells, secreted proteins etc), the physical properties for each component and the stream flow rate.

Also required is a UnitModel class for each type of unit operation which the user wishes to consider during the synthesis procedure. UnitModel classes contain the physical properties screening steps along with design equations which are used to determine the composition, flow and number of streams leaving each unit. In addition, capital and operating costs are estimated within UnitModel classes. A detailed discussion of the bioprocess unit design and cost estimation techniques used in this work is presented later.

To convert the synthesis problem from a mixed integer non-linear program (MINLP) into a discrete optimisation problem the Jacaranda package converts all continuous variables into discrete values. In bioprocess synthesis, there are two types of continuous variables which must be discretised:

- *stream characteristics* such as components flow rates. The discretisation level (or base flow) for these variables may be chosen by the user. For example, the base flow rate for water may be set to 5 kg/hr. This means that the mass flow of water will be rounded to the nearest multiple of 5 for each stream created during a synthesis run.
- *unit design parameters* such as the operating pH in an ion-exchange column. Design parameters may be considered as degrees of freedom during synthesis and design. A range and discretisation level for design parameters can also be specified by the user. For example, the pH in an ion-exchange column may be allowed to vary from 4 to 8 in steps of 2. The synthesis method designs the unit at each of the discrete values and evaluates each design.

The Jacaranda algorithm uses implicit enumeration to simultaneously generate and search all possible flowsheets in the discrete space in a depth first manner. To begin with, the feed stream is passed to each unit model and if a unit can process the stream then the composition and state of any stream(s) leaving the unit are calculated. From there the same procedure is repeated for each new stream. The algorithm continues until units which do not produce any streams are encountered. Such terminating units, or product tanks, are defined so as only to accept streams which contain a product at the desired purity. Computational efficiency is achieved in two ways:

1. *Dynamic programming*: Due to the discrete nature of the synthesis methodology it is possible for streams with the same composition and properties to arise via different synthesis routes. When stream characteristics are identical, the flowsheeting possibilities for processing the stream are also identical. In the Jacaranda process synthesis

package computational effort is reduced by identifying situations in which identical streams arise, and synthesising the corresponding flowsheets only once. Of course we can only check whether streams have the same composition in a discrete sense. To do this the Stream class must have a unique 'key' string for identification purposes (Fraga, 1996b).

2. *Cost pruning*: branches in the search graph may be pruned if their cost exceeds that of a different alternative before the branch has been exhaustively searched.

We now discuss the bioprocess unit design and costing methods in detail.

2.1 Unit Design Techniques

The bioprocess unit design procedure consists of a preliminary screening procedure followed by the actual design calculations which are used to determine effluent stream characteristics and cost information. The design procedure is summarised in Figure 1.

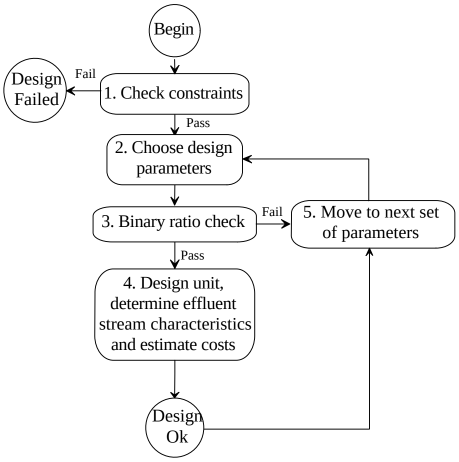
Begin
Design Failed
Fail
1. Check constraints
Pass
2. Choose design parameters
3. Binary ratio check
Fail
5. Move to next set of parameters
Pass
4. Design unit, determine effluent stream characteristics and estimate costs
Design Ok

Figure 1: The bioprocess unit screening and design procedure

In general, bioprocesses can be considered to consist of two subprocesses: fermentation and downstream purification (Lienqueo *et al.*, 1996). In this paper we focus upon the downstream purification section of the process. The effects of fermenter design issues on bioprocess synthesis have been investigated elsewhere (Steffens *et al.*, 1999).

2.1.1 Screening Units During Synthesis

The bioprocess synthesis algorithm developed in this work uses physical properties information to screen units thereby reducing the search space. This is advantageous because bioprocesses usually involve non-sharp separations and have many different available technologies which makes the search space large. Two types of tests are used to eliminate units which are not feasible for a particular separation:

1. *Design constraints* which are physical limits on pieces of equipment. For example, packed bed chromatography columns cannot process streams which contain solids. Table 1 summarises the design constraints used in this work.

2. The *binary ratio* check which utilises the concept of a key property, or driving force, which each unit exploits to achieve separation. This is discussed in more detail below.

Table 1: Bioprocess unit design constraints

Unit	Constraints[†]	Source
Ultrafilter	$1\mu m > x > 0.001\mu m$ $c_f < 200g/L$ $c_r < 500g/L$	Asenjo (1990) pp 210
Microfilter	$10\mu m > x > 0.1\mu m$ $c_{rs} < 500g/L$	Asenjo (1990) pp 210 Ho and Sirkar (1992) pp577
Diafilter	$1 > x > 0.001\mu m$ $c_f < 400g/L$	Asenjo (1990) pp 210
Centrifuge	$20 < c_{fs} < 150g/L$	Kennedy and Cabral (1993) pp 169
Rotary drum filter	$200\mu m > x > 1\mu m$ $c_{fs} < 70\% \; w/w$	Gabler (1985) Kennedy and Cabral (1993) pp 87
Chromatography column	$c_f < 70g/L$ No solids in the feed	Leser and Asenjo (1992)
Homogeniser	$c_{fs} < 750g/L$	Wheelwright (1991) pp64

The use of physical properties information for bioprocess synthesis is not a new concept. In the downstream processing design methodology described by Wheelwright (1987), it is recommended that the engineer selects units which exploit the greatest differences in physical properties between components. Leser and Asenjo (1992) go a step further and define a separation coefficient which is a function of the physical properties difference between components and may be used to choose between high resolution purification operations.

In this work we use a method developed by Jaksland *et al.* (1995) for chemical process synthesis to screen out infeasible unit operations for bioprocesses. Each separation process, whether chemical or biochemical, exploits specific property differences to facilitate purification of the various components in a stream. In this synthesis approach the key driving force, and corresponding property, utilised by each technology is identified (Table 2). To determine the applicability of any particular unit operation for the separation of a pair of components the ratio of this property (binary ratio) is calculated for each pair. If it is large enough then the unit is considered feasible for the separation.

A simple test is used to identify candidate separation operations by comparing two numbers:

1. *Binary ratios*: the potential driving force for the separation of any two components is quantified by calculating the ratio of the physical property governing the separation in the unit being considered (binary ratio).

[†] See §6 for symbol definitions

Table 2: Physical properties used by bioprocess unit operations

Unit Operation	Physical property	n	Source
Centrifugation	Settling velocity	1.1	
Conventional filtration	Size (particle)	1.1	
Rotary drum filtration	Size (particle)	1.1	
Microfiltration	Size (particle)	3.0	Jaksland *et al.* (1995)
Ultrafiltration	Size (molecular)	3.0	Jaksland *et al.* (1995)
Diafiltration	Size (molecular)	3.0	Jaksland *et al.* (1995)
Ion exchange	Net charge	2.5	
Gel chromatography	log(Mol. Wt.)	1.05	
Hydrophobic Int. Chr.	Hydrophobicity	1.5	

2. *Feasibility indices*: the binary ratios are then compared to feasibility indices, n, which are predefined constants for each separation technology. The indices define how large the binary ratio must be before a separation is feasible. Suggested values for n are shown in Table 2. Where sources are omitted recommended values were based on an analysis of the sensitivity of the synthesis results to the feasibility indices. Details of the investigation, which was performed using the first case study system, are presented below.

In combination, the design constraints and physical properties screening techniques reduce the problem size thereby making the synthesis problem simpler to solve without eliminating economically desirable processes. When a candidate unit passes the screening tests for a particular separation we proceed with the design and costing calculations for that unit. These are now described in detail.

2.1.2 Design and Cost Estimation Techniques

Two different approaches are used to determine the effluent stream characteristics for a separation unit depending upon the unit type: splitting or fractionating units. The former operate in a more traditional chemical engineering sense, that is by splitting the incoming stream into two with significantly different composition, hence achieving separation. The latter, however, generally operate batchwise and produce several fractions or cuts. Chromatography columns, where the various components are eluted sequentially, are a particularly common example of a fractionating unit.

Stream Splitting Units

Units which operate in a stream splitting sense are designed using a similar concept to that used in shortcut distillation design. The design procedure is based on the assumption that each unit operation exploits a particular physical property to achieve separation (Table 2). Initially, a list of the feed components, ordered in terms of this physical property, is generated. Two 'key' components are selected as an adjacent pair from this list. Prior to any design calculations, the binary ratio of the two key components is calculated and compared with the feasibility index for the unit under consideration. If the binary ratio is greater than the feasibility index, the design calculations are performed.

To calculate the effluent stream compositions, we consider the upper key and any components higher in the list as one. Similarly for the lower key and the components below it. Design assumptions and mass balances are then used to calculate the effluent stream compositions. For each unit, the design procedure is repeated at most n-1 times, where n is the number of components in the feed to the unit. Each design is performed using a different pair of adjacent components as the keys. However, before a unit is designed the screening procedure, discussed above, must be passed.

As an example, consider the design of an ultrafiltration unit where the feed components are sorted according to size (molecular or particle). The unit is designed n-1 times with each adjacent pair of components in the feed stream as the keys. For each design, the large key and any larger components are assumed to be impermeable (i.e. 100% rejected) while the small key and any smaller components are completely permeable (0% rejected). The composition of the effluent streams are then estimated using mass balances. Design assumptions and cost information for the various stream splitting unit operations considered in this paper are now summarised:

Ultrafilter: The small key is assumed to be completely permeable and the large key impermeable. The concentration factor, CF, is specified as a design parameter. Membrane area is estimated by calculating the limiting flux using a concentration polarization model (Ho and Sirkar, 1992: pp401-403). A gel concentration, c_g, and mass transfer coefficient, k, are assumed (Appendix A). We also assume operating costs are dominated by membrane replacement (once per year) and energy costs (Appendix B).

Microfilter: A microfiltration unit is designed in the same way as an ultrafilter. Cake resistance is assumed to be dominant and a concentration polarization model is used to estimate the flux and calculate membrane area (Ho and Sirkar, 1992: pp485). Operating costs are assumed to consist of energy and membrane replacement costs only.

Diafilter: As for the Ultrafilter except the dilution factor, D_F, is specified as a design variable instead of CF.

Centrifuge: The keys are chosen from the feed solid components which are ranked according to settling velocity (Wheelwright, 1991: pp 76). The upper key and any components with a higher settling velocity completely distribute into the slurry stream while the lower key and slower components move into the less dense stream. Soluble components are assumed to uniformly distribute between the two effluent streams. The slurry solids concentration, c_s, is specified as a design parameter (Appendix A). Centrifuge capacity is calculated using the design equations given in Brunner and Hemfort (1988) and parameter values are summarised in Appendix A. For costing purposes a nozzle discharge disc stack centrifuge is assumed. Operating costs are assumed to consist of energy expenses only (Appendix B).

Rotary Drum: As for the centrifuge the keys are selected from the solid components in the feed which are sorted in terms of particle size. The large key and larger components are assumed to be completely rejected by the filter while the small key and smaller components pass through the filter. The wash rate (w = volume wash water per volume slurry) and cake solids concentration, c_k, are both design parameters (Appendix A). The amount of solute remaining in the filter cake is estimated as a function of the wash rate (Kennedy and Cabral, 1993: pp 90). To estimate the filter area we assume a constant specific cake resistance (Belter *et al.*, 1988). For parameter values see Appendix A. Operating costs include energy and filter aid expenses only (Appendix B).

Fractionating Units

Chromatography columns, which fractionate rather than split, cannot be designed using the key component technique. The elution profile must be estimated so that the amount of each contaminant which leaves with the desired product can be estimated.

As was the case for the stream splitting units, we assume each high resolution purification unit exploits a particular physical property to achieve separation (Table 2). The feed components are sorted in descending order according to this physical property. Because chromatography columns usually target a particular protein (i.e. the product) we calculate the binary ratios with respect to the product protein. The binary ratios are calculated for all components and then summed using the contaminant mass fractions as weighting coefficients. If this average binary ratio is above n (Table 2), the separation is considered feasible and the column is designed.

A bioprocess stream contains many different types of complex biological compounds which interact with each other and with the column making the system difficult to model (Leser *et al.*, 1996). For this reason an empirical approach which estimates the effluent composition for a chromatography column using physical properties differences is used here (Leser *et al.*, 1996). An estimate of the peak width and the deviation factor are used to calculate the contaminant concentration in the purified stream as a function of its feed concentration. The deviation factor is defined as the physical property difference between the contaminant compound and the product of interest. The approach has been developed for ion-exchange, gel filtration and hydrophobic interaction chromatography columns and design assumptions and cost information for these units is as follows:

Ion exchange column: Feed components are sorted according to their net charge which is also used to calculate the binary ratios. The charge density of a protein, as defined by Leser *et al.* (1996), is used to calculate the deviation factors for the contaminant proteins. The column volume is estimated by assuming a binding capacity, B_c, and residence time, t_r (Appendix A). Components with opposite charge to the resin are assumed to completely bind while those with the same charge do not bind at all. Capital and operating costs were taken from data generated using the BioPro Designer software (Intelligen, 1997).

Gel filtration column: Log of the molecular weight (Leser *et al.*, 1996) is used to sort the feed components, calculate the binary ratios and calculate the deviation factor for a gel filtration column. Column volume is determined by specifying the sample volume, B_{sam} (% of column volume), and the residence time, t_r (Appendix A). Capital and operating information were from the same source as for the ion exchange column.

Hydrophobic interaction column: Hydrophobicity, as defined by Leser *et al.* (1996), is the physical property used to design a hydrophobic interaction chromatography column. Column volume is calculated in the same way as for ion exchange except that the binding fraction, F_i, for each component is assumed to have a linear relationship with the hydrophobicity, ϕ. Cost information is also from Intelligen (1997).

In addition to the separation units described thus far there are two important processes which do not separate components and, therefore, do not fit into either category: homogenisation and renaturation. Their design and cost assumptions are now discussed:

Homogeniser: We assume a high pressure homogeniser and specify the percentage recovery (Appendix A). The average size of the cell debris is also specified. Operating costs consist of energy expenditure only (Appendix B).

Renaturing tank: When the product protein forms inclusion bodies a solubilisation and renaturing tank is required. This unit is designed by specifying a residence time, t_r, and yield, y (Appendix A). The incoming stream is diluted with water to achieve a specified protein concentration (Appendix A). Capital costs are estimated as for a stirred tank and chemical costs are assumed to be the major operating expense (Appendix B).

In addition to capital and operating costs, we must estimate the value of the product and balance this against expenses. A logical way to do this is to use a negative cost (i.e. revenue) for any product which meets the required specifications. However, a cost scheme which incorporates solely positive cost data is advantageous to make best use of cost pruning in the synthesis algorithm. To achieve this we have used the following scheme: zero cost for any product leaving the process which meets specifications; and a positive cost for product which does not (i.e. wasted product). This cost scheme drives the synthesis process towards the production of on-specification product without using negative costs.

2.2 Enhancements to Jacaranda

The Jacaranda system provides an extensible framework for process synthesis. Through the definition of appropriate Stream and UnitModel classes, the search procedure is extended to include screening based on physical properties. However, there are two aspects of Jacaranda which have been enhanced to cater for the special properties of biochemical process synthesis:

1. The large size of the search space makes it worthwhile to introduce an extra pruning operation.
2. The ability to generate a ranked list of N solutions, instead of just the optimal flowsheet, has been enhanced to increase the diversity of solutions presented for any given value of N.

These two extensions are now described in more detail.

2.2.1 Heuristic Leaf Pruning

At each node in the search graph, the system considers each of the available units in turn. If a unit can handle the stream associated with the node, the set of discrete alternative designs are then processed. The outputs of each alternative become new nodes in the recursive search procedure. Because the search only backtracks when a node can only be processed by units which are final product or waste tanks, the depth of the search graph can be arbitrarily large when non-sharp separations are allowed. The ability to selectively but automatically prune some of the branches in the search procedure is necessary in these cases to ensure that solutions can be found in reasonable time.

The system has been extended to allow the depth-first search to terminate when a stream is found to match the criteria associated with a product or waste tank. Instead of considering all units that can handle a given stream, the order of the units in the problem definition becomes a heuristic which can eliminate unnecessary searching. For example, if units are ordered so that the main product tank comes first, any stream which is a valid product will not be processed further.

Although the addition of this heuristic to the search procedure reduces the search space, it does so fully under user control. Only those product or waste tank units which have strict criteria will typically be placed at the beginning of the list of units. In Section 3.1, we use a case study to quantify the reduction in the size of the search space which can be achieved using the leaf prune feature.

2.2.2 Enhancing N-best diversity

The discrete nature of the implicit enumeration search allows us to generate not just the optimal solution but a ranked list of the N-best solutions, for any specified value of N. For small values, this list is generated with little extra computational effort compared to identifying the optimal solution. For example, Figure 2 shows the relationship between the computational time and N for the first case study, the details of which are presented below.

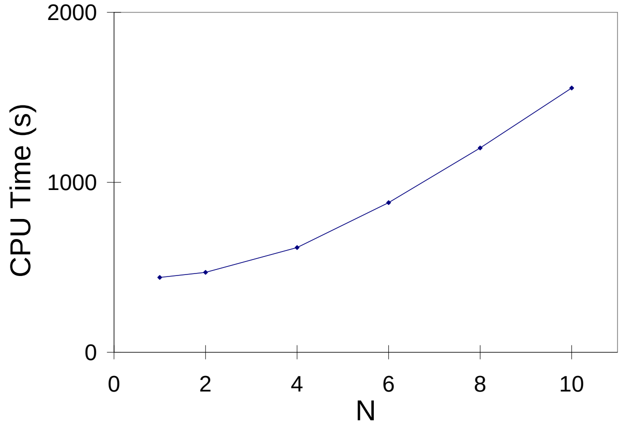

Figure 2: An example showing the effect of N on CPU time (on an IBM RS6000)

The list of solutions generated by the synthesis algorithm should differ in aspects which are of interest to the designer. For example, generating several solutions which have identical structure but differ in some operating condition may not provide enough diversity for the design engineer.

For biochemical processes, this diversity may not be enough due to the variety of processing steps, the degree of non-sharpness and uncertainty in cost models. For example, consider a stream with three components, ABC, of which A is the desired product and B and C are waste products. Figure 3 shows two possible solutions to the synthesis problem (note: aBC denotes a stream containing a small amount of A and significant amounts of B and C).

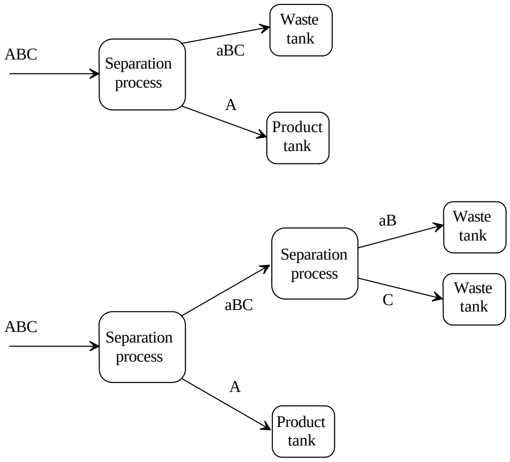

Figure 3: An example showing lack of solution diversity

For the engineer, there is really only one interesting feature which is common to both solutions: the product A is separated from the initial stream first. However, it is possible that both of the structures presented in Figure 3 are cheaper than any more interesting alternatives which would not be presented if the engineer had only asked for the two best solutions.

To increase the diversity of flowsheets generated by the synthesis algorithm we have extended the procedure for encoding of solution structures, which is used to compare solutions, to include the concept of interesting product nodes. Any structure which does not lead to an interesting product, as defined by the user, is encoded simply as "...". In the example above, if only a product tank is marked as interesting the two solutions in Figure 3 have the same encoding which means a different alternative would be ranked second thereby increasing the diversity.

3 Results

Two case studies have been chosen to illustrate the use of the synthesis algorithm and the new BioStream and UnitModel classes. Firstly we wish to synthesise a downstream flowsheet for a process where a protein which is secreted from *S. Cerevisae* is to be purified. In contrast, the second case study involves the purification of bovine somatotropin (BST), a product which forms inclusion bodies within the *E. coli* host cells. We now discuss the case study details and synthesis results in detail.

3.1 A Secreted Protein

Although many genetically engineered protein products are produced intracellularly, a significant fraction are secreted through the cell wall into the growth medium by the host organism. This represents a significant advantage because subsequent recovery and purification steps are often simple in comparison with those required for intracellular products (Wheelwright, 1991). Indeed, many researchers have attempted to genetically induce the secretion of protein products for this reason (Klein *et al.*, 1991; Nilsson, 1990). The first case study represents a simple version of one such process where a product is produced and secreted by *S. Cerevisae* (Baker's yeast) cells. The feed stream composition and physical property data used for this case study are presented in Table 3 and in Appendix C. We assume two contaminant proteins (protein 1 and 2 in Table 3) are secreted into the medium with the product.

Component	c_i	Size	Density	ϕ	MW	Base flow
	(g/L)	(μm)	(g/L)		(g/mol)	(kg/hr)
1. Water	970	4.0×10^{-4}	1000		18	50.0
2. *S. Cerevisae*	60	4.0	1100			10.0
3. SO_4^-	0.25	4.0×10^{-4}	1050	0.01	96	0.1
4. NH_4^+	0.10	2.0×10^{-4}	1050	0.01	18	0.1
5. Antifoam	0.15	1.0×10^{-2}	985	1.00	500	0.1
6. Glucose	0.05	1.0×10^{-3}	1250	0.10	180	0.1
7. Protein 1	1.25	2.0×10^{-3}	1300	0.71	18500	0.1
8. Protein 2	1.50	5.0×10^{-3}	1300	0.36	170000	0.1
9. Product	12.00	8.0×10^{-3}	1300	0.90	250000	1.0

Table 3: Feed stream characteristics for the secreted protein case study

Eight unit operations including ultrafiltration, diafiltration, microfiltration, centrifugation, rotary drum filtration, ion exchange chromatography, gel filtration and hydrophobic interaction chromatography were considered for this synthesis problem. In addition we have defined a product tank which only accepts a stream containing the product with less than 0.1% contaminants and concentration greater than 100 g/L. The product was valued at 1000 $\$/kg$ and annual production is 50 tonnes per annum. Figure 4 shows the top two purification flowsheets generated by the synthesis software using the feasibility indices given in Table 2.

The flowsheet, in particular the recovery section, is relatively simple with only five downstream unit operations required to purify the product. This characteristic was common to all of the flowsheets examined by the authors (ten best) and confirms the significant advantages associated with secreted protein systems (Wheelwright, 1991).

Of the first ten flowsheets generated by the synthesis package most of the variation was in the purification section (i.e. order and type of chromatography columns). All of the ten best flowsheets use at least one hydrophobic interaction column. This is an expected result as the product's hydrophobicity is significantly different from that of the other soluble contaminants (Table 3) and the synthesis method, therefore, recommends that we exploit this

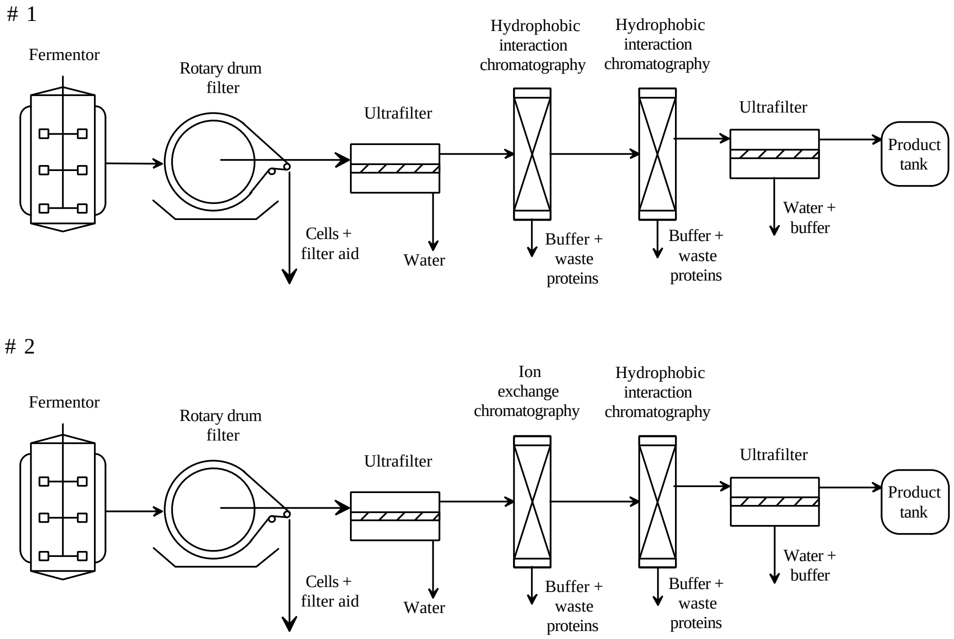

Figure 4: Best two flowsheets identified by the synthesis software

difference. Ion exchange columns also featured in many of the flowsheets generated by the synthesis software.

The alternatives considered by the synthesis software for cell removal included rotary drum filtration, centrifugation and cross flow filtration. In the biotechnology industry this decision is a contentious one at present as all three technologies have associated advantages and disadvantages. In particular, the analysis presented by Tutunjian (1986) showed that cross flow filters have lower capital and operating costs when separating streams containing smaller bacterial cells (e.g. *E. coli*). However, for larger cells, as is the case here, centrifuges and rotary drum filters are capable of a higher throughput and, therefore, become more economical. The synthesis procedure chose rotary drum filtration for cell removal in all of the ten best flowsheets because it had a higher predicted product recovery than centrifugation and was more economical than cross flow filtration for this case study.

It is useful to examine several statistics of the synthesis problem including:

1. The percentage of units which were screened due to physical constraints. This statistic represents the number of times all units operations were considered infeasible because a physical constraint was not satisfied, divided by the total number of times the units were considered.

2. The percentage of units which were screened using binary ratios out of those which passed physical constraint screening (Step 1 in Figure 1).

3. The number of unique synthesis problems (i.e. unique streams) which is indicative of the superstructure size.

4. The number of repeated problems which did not have to be solved because they had already been encountered during the synthesis. This tells us whether the dynamic programming aspect of the algorithm is well suited for downstream purification synthesis problems.

5. The number of cost pruning operations.

6. The maximum search depth.

Table 4 compares the statistics for the protein A purification synthesis problem when run with and without the cost and leaf pruning features. The following observations pertaining to the statistics can be made:

- Both types of screening reduced the size of the superstructure significantly with approximately half of the units considered infeasible due to physical constraints and another 60% of those which passed the first stage (Figure 1) screened using binary ratios (Step 3 in Figure 1).
- On comparing columns 1 with 2 and 3 with 4 (Table 4) we can conclude that the introduction of cost pruning had a relatively small effect on the number of unique problems (i.e. problem size) and the CPU time used to solve the problem.
- Leaf pruning reduces both superstructure size and CPU time significantly and represents an important enhancement to the synthesis algorithm (i.e. compare columns 1 and 3).
- The number of repeated problems is at least 3 times the number of unique problems in all four situations showing that the dynamic programming approach is well suited to downstream process synthesis.

Statistic	Leaf and cost pruning	Leaf pruning only	Cost pruning only	No pruning
Units screened via constraints(%)	53	52	56	55
Units screened via binary ratios(%)	67	66	72	62
Number of unique problems	7063	9187	18237	22969
Number of repeated problems	17266	27153	72528	101109
Number of cost pruning operations	1783	0	4321	0
Maximum search depth	16	16	18	18
CPU time (minutes on an IBM RS6000)	13.1	17.4	55.5	70.7

Table 4: Statistics for the protein A synthesis problem (N=10)

The discretisation level used to map stream composition and properties into the discrete space is crucial to the success of the synthesis algorithm. Too coarse a setting may result in unrealistic synthesis results while a fine discretisation can cause a dramatic increase in the size of the search space. Table 5 shows the effect of the discretisation level of the soluble contaminant compounds (components 3-8 in Table 3) on the synthesis results.

Discretisation scheme	Base flow (kg/hr)	Number of unique problems
1	1.0	1571
2	0.5	2387
3	0.1	6880
4	0.05	8828

Table 5: Effects of the discretisation level on synthesis results ($N = 10$)

As was expected the size of the search space, as indicated by the number of unique problems, increases significantly when a finer discretisation level is used (i.e. smaller base flow). In terms of synthesis results schemes 3 and 4 were identical (for N=10) while for scheme 2, only flowsheets 9 and 10 differed. The flowsheets presented in Figure 4 correspond to discretisation scheme 3 and all use two chromatography columns to purify the product protein. Results for scheme 1, however, reveal that the discretisation level is too coarse because the 10 best flowsheets all use only one chromatography column, in contrast to the flowsheets generated using the three finer discretisation levels. This is because the discretisation method removes any component with a flow that maps to zero (i.e. those with a flow below half of the base flow) from the stream under consideration. Thus the synthesis algorithm incorrectly predicts that fewer units are required to achieve the desired end product. From this we can conclude that it is important to study the effect of the discretisation level on a case by case basis to ensure accurate results and minimal computational resources are used. For this case study, for example, a discretisation level at least as fine as that used for scheme 2 in Table 5 should be used.

In addition to the discretisation level, the feasibility indices (n) also have an effect on the size of the search space and, perhaps, the synthesis results. For this reason we have examined the sensitivity of the synthesis algorithm to the feasibility indices for the various unit operations. In particular we found that the synthesis results were insensitive to n for microfiltration, centrifugation and rotary drum filtration. This is because all of these units perform solid/liquid separation tasks which are relatively few when compared to liquid/liquid separations in this case study. However, values for ultrafiltration, diafiltration, ion exchange, gel filtration and hydrophobic interaction chromatography all had a significant effect on the size of the search space and, in some cases, on the synthesis results. For example, in Figure 5 we show the effect of n for ultrafiltration on the number of unique problems, an indicator of the size of the search space.

When n was increased above 2.5 the search space was significantly reduced leading to lower computation times. The 10 best flowsheets were identical for all values of n up to and including 3. However, when n was further increased a different set of optimal flowsheets resulted because the binary ratio screening resulted in the elimination of several flowsheets which were originally in the top 10. We would like to set n so that the search space is reduced without eliminating important solutions. Based on the above analysis a value of 3 for ultrafiltration is appropriate in this case. Similar analyses for other unit operations resulted in the values for n presented in Table 2. We now introduce the features of the second case study and present the synthesis results.

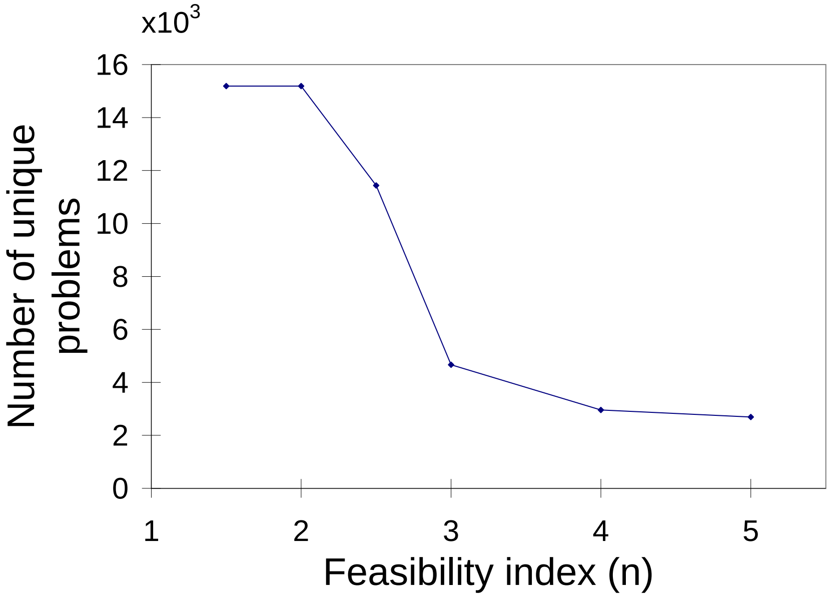

Figure 5: The effect of n for ultrafiltration on the search space size

3.2 Bovine Somatotropin

Bovine somatotropin (BST), also known as bovine growth hormone, is an animal product produced in recombinant *E. coli*. BST is of particular interest for use in agricultural applications as it enhances milk production in cows. The *E. coli* fermentation of BST produces an intracellular product in the form of inclusion bodies. The isolation of active proteins from inclusion bodies requires complete denaturation followed by chemical refolding. The downstream process must, therefore, involve cell disruption, solubilisation of inclusion bodies and protein refolding. Table 6 summarises the characteristics of the fermenter effluent stream which is used to demonstrate the synthesis technique in this work. Further physical properties data including net charge is presented in Appendix D. Annual production of BST is approximately 2 t/y.

Component	c_i	Size	Density	ϕ	MW	Base flow
	(g/L)	(μm)	(g/L)		(g/mol)	(kg/hr)
Water	950.0	4.0×10^{-4}	1000		18	5.0
E. coli	60.0	1.5	1100			1.0
SO_4^-	0.1	1.0×10^{-3}	1050	0.01	96	0.5
NH_4^+	0.1	5.0×10^{-4}	1050	0.01	18	0.5
Antifoam	0.3	1.0×10^{-1}	985	1.00	500	0.5
Glucose	1.5	1.0×10^{-3}	1250	0.10	180	0.5

Table 6: Feed stream characteristics for the BST case study

We did not have to consider the cell contents in the first case study because the product was secreted. This is not the case for BST production, however, where the product is intracellular and cells must be disrupted. This means the cell contents must be modelled and to do this we make several simplifying assumptions:

- Cells are assumed to contain 13 dissolved proteins along with the inclusion bodies. These proteins were identified as the most abundant, apart from the product, in a typical *E. coli* cell and were subsequently characterised in work presented by Leser *et al.* (1996). Data for all biochemicals also include a titration curve (i.e. charge as a function of pH) which is used for ion exchange chromatography selection and design (Appendix E).
- Inclusion bodies are assumed to contain 50% BST with the remainder approximated as a single contaminating protein (Appendix D).
- Cell debris is assumed to consist of cell wall material only and the average particle size is specified as part of the homogeniser design procedure (Appendix A).

The composition and physical properties of the *E. coli* cells used for this case study is presented in Table 7.

Component	% w/w	ρ (g/L)	Size (μm)	Base Flow (kg/hr)
Water	70	1000	4.0×10^{-4}	Appendix E
Cell wall material	7	1150	0.65	
Soluble proteins	18	1300		
Inclusion bodies	5	1270	0.4	

Table 7: *E. coli* cell composition for the BST case study

Eleven different unit operations were considered in the synthesis procedure including: microfiltration, centrifugation, ultrafiltration, diafiltration, rotary drum filtration, homogenisation, denature/refolding tanks, cation exchange, anion exchange, hydrophobic interaction chromatography and gel filtration. Feasibility indices used for this case study were presented in Table 2. In addition we have defined two terminating units: a product tank which accepts streams containing BST at the desired concentration (300 g/L) and purity (99.9%) and a waste tank which accepts all other streams. The product was valued at 100 $\$/g$ (Bailey and Ollis, 1986). Figure 6 shows the optimal purification flowsheet found using the synthesis software.

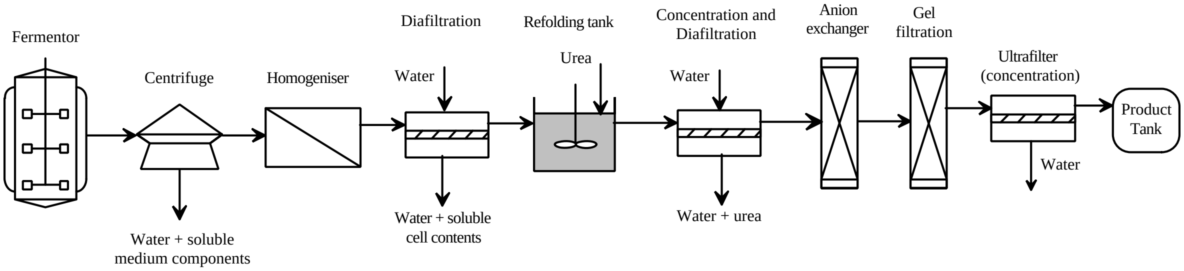

Figure 6: Optimal purification flowsheet for BST production

As is expected for an intracellular product the recovery section of the process is relatively complex and contains five units. Cells are first concentrated in a centrifuge which serves to reduce the volume which the homogeniser must process. After cell disruption a diafiltration unit removes the bulk of the soluble cell contents. Next the inclusion bodies are dissolved and the BST allowed to refold in the presence of urea which is then removed using a combination of concentration and diafiltration. From there the stream is ready to go into the high resolution purification part of the plant which consists of two types of chromatography, anion exchange and gel filtration, before concentration using an ultrafiltration unit. To check the validity of the flowsheet generated by the synthesis package we compare it with an industrial scale BST flowsheet which is presented in Figure 7 (Wheelwright, 1991).

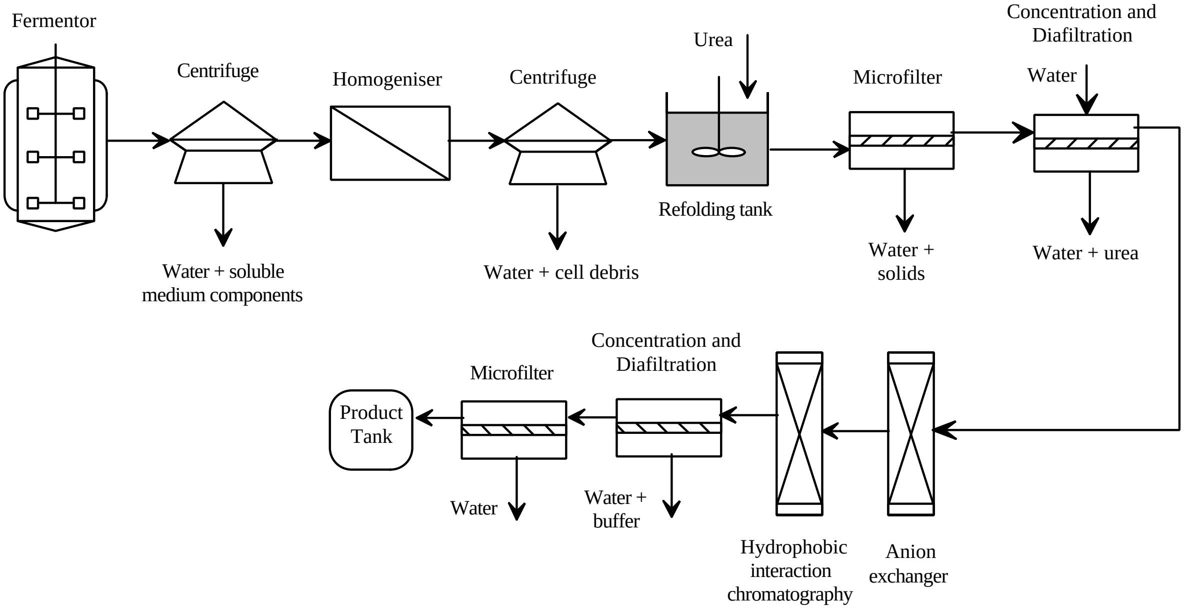

Figure 7: Industrial purification flowsheet for BST production

While it can be seen that the basic structure of the two flowsheets is similar the industrial scale system is slightly more complex with two extra units. The product recovery section in the industrial process uses two centrifuges, before and after homogenisation, whereas the optimal flowsheet generated by Jacaranda recommends an ultrafiltration unit after cell disruption. As was previously discussed the choice between centrifugation and cross flow filtration for cell concentration and debris removal is a difficult one which has to be investigated in more detail for each case. Statistics for the BST synthesis problem are listed in Table 8.

The complexity of the synthesis problem, as indicated by the number of unique problems and the maximum search depth, is significantly higher than for the first case study as expected. As with the protein A synthesis problem screening is effective in reducing the superstructure size with both types of screening eliminating greater than half of the possible units. Cost pruning is also relatively ineffective for the BST synthesis problem. Finally we note that the time required to solve this complex synthesis problem is reasonable (15 minutes on an IBM RS6000) making the algorithm suitable for use in an industrial setting.

Statistic	Value
Units screened via constraints(%)	
Units screened via binary ratios(%)	
Number of unique problems	
Number of repeated problems	
Number of cost pruning operations	
Maximum search depth	
CPU time (minutes on an IBM RS6000)	

Table 8: Statistics for the BST synthesis problem (N=5)

4 Conclusions

In this paper we have developed and presented a synthesis technique for generating optimal downstream processing flowsheets for the biotechnology industry. The technique integrates the idea of screening units via physical properties information into the Jacaranda implicit enumeration synthesis tool. Screening is an important enhancement to the technique for solving bioprocess synthesis problems which are often characterised by a large number of components and a wide selection of unit operations. Further refinements to the algorithm include leaf pruning, which further reduces the size of the search space, and an option for enhancing the diversity of flowsheets generated by the package.

Two case studies were used to evaluate and validate the performance of the synthesis algorithm: a secreted protein and bovine somatotropin (BST). The results show that the algorithm is well suited for solving bioprocess synthesis problems. Enhancements to the original synthesis algorithm (i.e. screening and leaf pruning) significantly reduced the size of the search space thereby improving computational efficiency. The case study results show that we have developed a useful synthesis tool for use in the biotechnology industry where the evolution of new processes is a relatively common and expensive procedure.

5 Acknowledgements

The work was carried out as part of the Centre for Process Systems Engineering and financial support from the Engineering and Physical Sciences Research Council is gratefully acknowledged.

6 Nomenclature

ρ = Density (g/L)
x = Particle or molecule size (μm)
pI = Isoelectric point
MW = Molecular weight (g/mol)
c_k = Cake solids concentration (g/L)
c_{fs} = Feed solids concentration (g/L)
c_{rs} = Reject solids concentration (g/L)
c_g = Gel concentration (% w/w)
t_r = Residence time (h)
n = Feasibility index
F_i = Binding fraction
N = Number of synthesis solutions
ϕ = Hydrophobicity as defined by Leser *et al.* (1996)
CF = Concentration factor (m^3 feed/m^3 reject)
D_F = Dilution factor (m^3 water/m^3 feed)
c_i = Concentration of component i (g/L)
c_s = Slurry solids concentration (g/L)
c_f = Total feed concentration(g/L)
w = Wash ratio (m^3 water/m^3 feed)
k = Mass tranfer coefficient ($m^3/m^2/s$)
y = Yield (g protein renatured/g total protein)
B_c = Binding Capacity (mg/mL)
B_{sam} = Sample volume (% column vol)

7 Bibliography

Asenjo, J. A. 1990. Separation processes in biotechnology. Marcel Dekker Inc., New York.

Asenjo, J. A., Herrera, L. and Byrne, B. 1989. Development of an expert system for selection and synthesis of protein purification processes. Journal of Bio/technology **11**: 275-298.

Bailey, J. E. and Ollis, D. F. 1986. Biochemical engineering fundamentals. 2nd ed. McGraw-Hill, New York.

Belter, P. A., Cussler, E. L. and Hu, W.-S. 1988. BioSeparations. Downstream processing for biotechnology. Wiley and sons, New York.

Brunner, K.H. and Hemfort, H. 1988. Centrifugal Separation in Biotechnological Processes, pp 1-50. In: A. Mizrahi (ed.), Downstream Processes: Equipment and Techniques, Alan R. Liss, Inc., New York.

Douglas, J. M. 1988. Conceptual Design of Chemical Processes. McGraw-Hill, USA.

Fraga, E.S. 1998. The Generation and Use of Partial Solutions in Process Synthesis. Transactions of the Institute of Chemical Engineers **76**(A1): 45-54.

Fraga, E. S. 1996a. The Automated Synthesis of Complex Reaction/Separation Processes Using Dynamic Programming. Transactions of the Institute of Chemical Engineers **74**(A): 249-260.

Fraga, E. S. 1996b. Discrete optimization using string encodings for the synthesis of complete chemical processes, pp. 627-651. In: C. A. Floudas and P. M. Pardalos (eds.), State of the Art in Global Optimization, Kluwer Academic Publishers, Dordrecht.

Friedler, F., Varga, J. B., Feher, E. and Fan, L. T. 1996. Combinatorily accelerated branch and bound method for solving the MIP model of process network synthesis, pp. 609-626. In: C. A. Floudas and P. M. Pardalos (eds.), State of the Art in Global Optimization: Computational Methods and Applications, Kluwer Academic Publishers, Dordrecht.

Gabler, F. R. 1985. Cell processing using tangential flow filtration, pp. 351-366. In: M. Moo-Young (eds.), Comprehensive Bio/technology: the principles, applications and regulations of Bio/technology in industry, agriculture and medicine, 1 ed., Pergamon, Oxford.

Grossman, I. E. and Kravanja, Z. 1995. Mixed Integer Non-linear Programming Techniques for Process Systems Engineering. Computers and chemical engineering **19**(Suppl.): S83-S88.

Ho, W. S. and Sirkar, K. S. 1992. Membrane Handbook. 1st ed. Van Nostrand Reinhold, New York.

Intelligen, Inc. 1997. BioPro Designer User Guide.

Jaksland, C. A., Gani, R. and Lien, K. M. 1995. Separation process design and synthesis based on thermodynamic insights. Chemical engineering science **50**(3): 511-530.

Kennedy, J. F. and Cabral, J. M. S. 1993. Recovery Processes for Biological Materials. John Wiley and Sons, Chichester.

Klein, B. K., Hill, S. R., Devine, C. S., Rowold, E., Smith, C. E., Galosy, S. and Olins, P. O. 1991. Secretion of active bovine somatotropin in Esherischia coli. Bio/technology **9**: 869-872.

Leser, E. W. and Asenjo, J. A. 1992. Rational Design of Purification Processes For Recombinant Proteins. Journal of Chromatography-Biomedical Applications **584**(1): 43-57.

Leser, E. W., Lienqueo, M. E. and Asenjo, J. A. 1996. Implementation in an Expert-System of a Selection Rationale For Purification Processes For Recombinant Proteins. Annals of the New York Academy of Sciences **782**: 441-455.

Lienqueo, M. E., Leser, E. W. and Asenjo, J. A. 1996. An Expert-System For the Selection and Synthesis of Multistep Protein Separation Processes. Computers and Chemical Engineering **20**(SA): S 189-S 194.

Nilsson, B. 1990. Fusions to Staphylococcal protein A. Methods in Enzymology **185**: 144.

Peters, M. S. and Timmerhaus, K. D. 1991. Plant design and economics for chemical engineers. 4th ed. McGraw Hill, New York.

Petrides, D. P. 1994. Biopro Designer - an Advanced Computing Environment For Modeling and Design of Integrated Biochemical Processes. Computers and Chemical Engineering **18**(SS): S621-S625.

Petrides, D. P., Calandranis, J. and Cooney, C. 1995. Computer-Aided-Design Techniques For Integrated Biochemical Processes. Genetic Engineering News **15**(16): 10.

Reisman, H. 1988. Economic analysis of fermentation processes. CRC Press, Boca Raton.

Smith, E. M. and Pantelides, C. C. 1995. Design of Reaction/Separation networks using detailed models. Computers and chemical engineering **19**(Suppl.): S83-S88.

Steffens, M. A., Fraga, E. S. and Bogle, I. D. L. 1999. Optimal System Wide Design for Bioprocesses. Bioprocesses. *To be presented at the 9th European Symposium on Computer Aided Process Engineering: ESCAPE-9*, Budapest.

Tutunijian, R. S. 1985. Cell separations with holllow fiber membranes, pp. 367-381. In: M. Moo-Young (eds.), Comprehensive Bio/technology: the principles, applications and regulations of Bio/technology in industry, agriculture and medicine, 1 ed., Pergamon, Oxford.

Wheelwright, S. M. 1987. Designing downstream processes for large-scale protein purification. Bio/technology **5**: 789.

Wheelwright, S. M. 1991. Protein purification: design and scale up of downstream processing. Hanser, Munich.

A Unit design parameters

Unit	Parameter values
Ultrafilter	$c_g = 25\%\ w/w$ $k = 1.0 \times 10^{-6}\ m^3/m^2/s$ $CF = 5.0\ m^3/m^3$
Microfilter	$c_g = 40\%\ w/w$ $k = 1.5 \times 10^{-5}\ m^3/m^2/s$ $CF = 5.0\ m^3/m^3$
Diafilter	$c_g = 25\%\ w/w$ $k = 1.0 \times 10^{-6}\ m^3/m^2/s$ $D_F = 3.0\ m^3/m^3$
Centrifuge	$c_s = 150\ g/L$ Bowl speed = 963 s^{-1} Number of discs = 72 Disc opening angle = 38^o Inner diameter = 0.072 m Outer diameter = 0.162 m Dynamic viscosity = 1.02 kg/ms
Rotary drum filter	$c_k = 35\%\ w/w$ $w = 1.0\ m^3\,water/m^3\,feed$ Cake resistance = $5.0 \times 10^{11}\ m/kg$ Pressure drop = 68 kPa Filtrate viscosity = 0.0011 $kg/m/s$ Cycle time = 180 s
Ion exchange chromatography	Maximum diameter = 1.0 m t_r = 5 h B_c = 20 mg/ml Column length = 0.25 m
Gel filtration	Maximum diameter = 1.0 m B_{sam} = 5 % Column length = 0.5 m
Hydrophobic interaction chromatography	Maximum diameter = 1.0 m B_c = 20 mg/mL t_r = 5 h Column length = 0.25 m
Homogenisation	Cell debris size = 0.65 μm Recovery = 99 %
Solubilisation and renaturing tank	t_r = 44 h y = 80 % Assume a solution of 3M guanidine HCl is used

B Capital and operating cost information

Unit	Capital costs	Operating costs	Source

Ultrafilter	1000 $\$/m^2$	Membrane = 250 $\$/m^2$
		Energy = 5 $kWh/m^3\ permeate$
		Energy costs = 0.04 $\$/kWh$
			Ho and Sirkar (1992) pp451

Microfilter	5000 $\$/m^2$	Membrane = 700 $\$/m^2$	Ho and Sirkar (1992) pp578
		Energy = 0.13 $\$/m^3\ feed$	Tutunijian (1985)

Diafilter	1000 $\$/m^2$	Membrane = 250 $\$/m^2$
		Energy = 5 $kWh/m^3\ permeate$
		Energy costs = 0.04 $\$/kWh$
			Ho and Sirkar (1992) pp451

Centrifuge	Reisman (1988) pp 64	Energy = 1.5 $kWh/m^3\ feed$ Energy costs = 0.04 $\$/kWh$	Kennedy and Cabral (1993) pp 169
Rotary drum filter	Peters and Timmerhaus (1991) pp 554	Filter aid = 5 $kg/m^3\ filtrate$ Filter aid cost = 0.33 $\$/kg$	Kennedy and Cabral (1993) pp 92
		Energy = 0.12 $\$/m^3\ feed$	Tutunijian (1985)
Ion exchange	Intelligen (1997)	Gel cost = 400 $\$/L$	Intelligen (1997)
chromatography			
Gel filtration	Intelligen (1997)	Gel cost = 300 $\$/L$	Intelligen (1997)
Hydrophobic interaction chromatography	Intelligen (1997)	Gel cost = 400 $\$/L$	Intelligen (1997)
Homogenisation	Reisman (1988) pp 67	Energy costs from Reisman (1988) pp 67	
Solubilisation and renaturing tank	Peters and Timmerhaus (1991) pp539	Chemical costs only (2.15 $\$/kg$ guanidine)	Asenjo (1990) pp779

C Net charge information for the first case study

Component	pH = 4	pH = 8	pH = 12
SO_4^-	-1.0	-1.0	-1.0
NH_4^+	1.0	0.0	0.0
Antifoam	0.0	0.0	0.0
Glucose	0.0	0.0	0.0
Product	1.3	0.2	-0.8
Protein 1	1.94	-2.5	-3.0
Protein 2	5.2	2.2	1.0

D Net charge information for the second case study

Component	pH = 4	pH = 6	pH = 8
SO_4^-	-1.0	-1.0	-1.0
NH_4^+	1.0	0.0	0.0
Antifoam	0.0	0.0	0.0
Glucose	0.0	0.0	0.0
BST	0.8	0.4	-0.1
Inclusion body contaminant protein	1.9	-0.5	-2.5

E Physical property information for the 13 main contaminant proteins in *E. coli* (Leser *et al.*, 1996)

Cell	% wt	Mol Wt	Charge	Charge	Charge	ϕ	Base Flow
protein			pH = 4	pH = 6	pH = 8		(kg/h)
1	16.1	18370	1.9	-1.8	-2.5	0.71	
2	10.1	85570	2.4	-2.8	-3.7	0.48	
3	6.6	53660	1.8	-0.5	-1.3	0.76	
4	7.9	120000	3.3	-1.1	-2.9	1.50	
5	6.9	203000	4.1	-1.9	-5.0	0.36	
6	3.6	69380	5.2	-1.9	-3.9	0.36	
7	11.0	48320	4.0	-1.4	-1.6	0.48	
8	9.7	93380	10.9	-0.8	-4.5	0.93	
9	10.7	69380	1.1	0.0	-0.3	1.00	
10	8.6	114450	10.4	0.6	-1.4	0.63	
11	5.6	198000	0.3	0.1	-1.0	0.06	
12	2.1	30400	5.2	1.5	0.2	1.00	
13	1.2	94670	11.7	2.7	0.8	1.00	
	100.0						