Journal of Pharmaceutics and Drug Research
JPDR, 3(2): 329-340
www.scitcentral.com

ISSN: 2640-6152

Original Research Article: Open Access

Optimization of Multiproduct Batch Plant Design for Protein Production Using Heuristics Algorithms

Youness El Hamzaoui[1*]

[*1]*Facultad de Ingeniería Universidad Autónoma del Carmen, Mexico.*

Received February 17, 2020; Accepted February 27, 2020; Published March 29, 2020

ABSTRACT

This work deals with the problem of the optimization of multiproduct batch plant design (MBPD) found in a chemical engineering process. The aim of this work is to minimize the investment cost and find out the number and size of parallel equipment units in each stage. For this purpose, it is proposed to solve the problem in two different ways: The first way is by using route-packing-based (RPBB) and the second way is by seed-based (SeedBB). This paper presents the effectiveness and performance comparison of RPBB and SeedBB for optimal design of multiproduct batch plant. The calculation results (investment cost, number and size of equipment, computational time, CPU time and idle times in plant) obtained by RPBB are better than SeedBB. This approach can facilitate the manufacturers of pharmaceutical drug to get an optimal design and makes up a remarkably suggested plan for having a benefit of efficient results.

Keywords: Mathematical modeling, Chemical engineering optimization, Route-packing-based algorithm, Seed-based algorithm, Batch plant design

INTRODUCTION

Pharmaceutical researchers and biotechnology companies are devoted to developing medicines, such as: Therapeutic proteins, human insulin, vaccines for hepatitis, food grade protein, chymosin detergent enzyme and cryophilic protease. This allows patients to live longer, heathier and more productive. However, in recent years, there has been an increased interest development of systematic method for the design of batch process in chemicals, food products and pharmaceutical industries. Basically, batch plants are composed of items operating in a discontinuous way, where each batch then visits a fixed number of equipment items, as required by a given synthesis sequence so called production recipe. Many works in the literature on batch process design are based on expressions that relate the batch sizes linearly with the equipment sizes [1-10]. The number required of volume and size of parallel equipment units in each stage is to be determined. Nevertheless, the design of batch plants requires involving how equipment may be utilized. However you look at it the optimal design of a multiproduct batch chemical process involves the production requirement of each product and the total production time available for all products has been considered. The number and size of parallel equipment units in each stage as well as the location and size of intermediate storage are to be determined in order to minimize the investment cost.

This paper proposes a solution to the optimal design of multiproduct batch plant design for protein production (MBPD). An integrated model was developed proposing two relevant heuristics to solve the MBPD's problem: route-packing-based-batch (RPBB) and seed-based-batch (SeedBB). The proposed heuristics are best for handling MBPD's enormous problem size with a large number of equipments.

We have found out that SeedBB performed effectively and gave a solution, but we would like to solve the problem more effectively, that's why we proposed to apply RPBB, an intelligent problem-solving method that has demonstrated its effectiveness in solving combinatorial optimization problem and satisfactory results have been obtained.

MATERIALS & METHODS

System description and experimental data

The case study, taken from the literature, is a multiproduct

Corresponding author: Youness El Hamzaoui, Universidad Autónoma del Carmen, Campêche, Mexico, Tel: 00521(999)9601138; E-mail: eyouness@pampano.unacar.mx, yelhamzaoui77@gmail.com

Citation: Hamzaoui YE (2020) Optimization of Multiproduct Batch Plant Design for Protein Production Using Heuristics Algorithms. J Pharm Drug Res, 3(2): 329-340.

batch plant for the production of proteins [11]. This example is used as a test bench since it provides models describing the unit operations involved in the process. The batch plant involves eight stages for producing four recombinant proteins, on one hand, two therapeutic proteins, human insulin (A) and vaccine for hepatitis (B) and, on the other hand, a food grade protein, chymosin (C) and a detergent enzyme, cryophilic protease (D). **Figure 1** is the flowsheet of the multiproduct batch plant considered in this study. All the proteins are produced as cells grow in the fermenter. It is hardly necessary to say that the number of intermediate storage tanks is an important constituent of our process: Three tanks have been selected: The first after the fermenter, the second after the first ultrafilter and the third after the second ultrafilter.

Vaccines and protease are considered to be intracellular. The first microfilter is used to concentrate the cell suspension, which is then sent to the homogenizer for the second microfilter, which is used to remove the cell debris from the solution proteins. The first ultrafiltration step is designed to concentrate the solution in order to minimize the extractor volume. In the liquid-liquid extractor, salt concentration (NaCl) is used as solution in order to minimize the extractor volume. In the liquid-liquid extractor, salt concentration (NaCl) is used to first drive the product to a poly-ethylene-glycol (PEG) phase and again into an aqueous saline solution in the back extraction. The second ultrafiltration is used again to concentrate the solution. The last stage is chromatography, during which selective binding is used to better separate the product of interest from the other proteins.

Insulin and chymosin are extracellular products. Proteins are separated from the cells in the first microfilter, where cells and some of the supernatant liquid stay behind to reduce the amount of valuable products lost in the retentate, extra water is added to the cell suspension. The homogenizer and the second microfilter for cell debris removal are not used when the product is extracellular. Nevertheless, the first ultrafilter is necessary to concentrate the dilute solution prior to extraction. The final step of extraction, second ultrafiltration and chromatography are common to both the extracellular and intracellular products. In **Table 1**, we make an estimation of production targets and product prices [12-14].

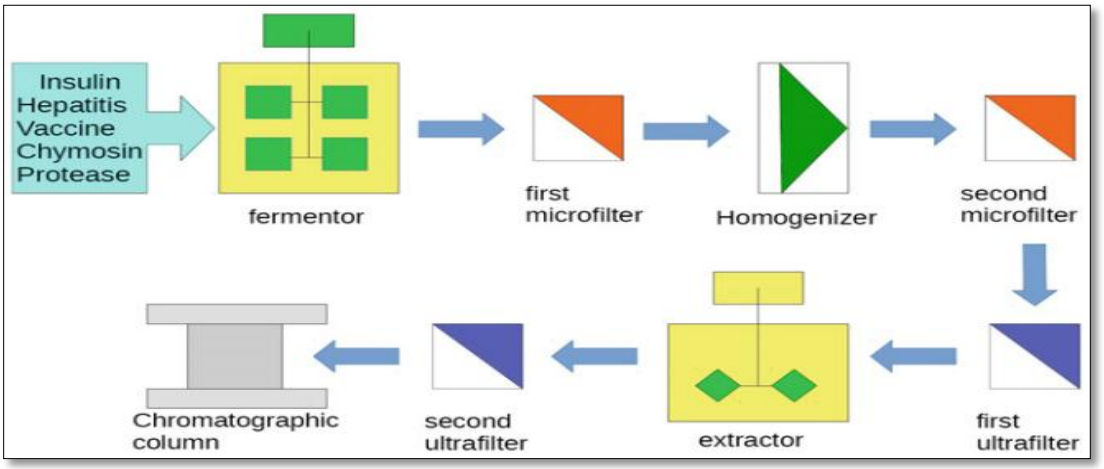

Figure 1. Multiproduct batch plant for protein production.

Table 1. Product prices and demands.

Product	Name	Production (kg/year)	Price (dollar/kg)
1	Insulin	1500	8000
2	Vaccine	1000	7500
3	Chymosin	3000	1000
4	Protease	6000	500

Problem statement

The model formulation for DMBP's problem approach adopted in this section is based on Montagna et al. [15]. It considers not only treatment in batch steps, which usually appear in all types of formulation, but also represents semi continuous units that are part of the whole process (pumps, heat exchangers, etc.). A semi-continuous unit is defined as a continuous unit alternating idle times and normal activity periods. Besides, this formulation takes into account mid-term intermediate storage tanks, the obligatory mass balance at the intermediate storage stage, which is one of the most efficient strategies to decouple bottlenecks in batch plant design. They are just used to divide the whole process into subprocesses in order to store an amount of materials corresponding to the difference of each sub-process productivity. In this section we describe the unit models from a conceptual standpoint and also the procedure to derive the data needed for solving the mathematical model. These data are summarized in **Tables 2 and 3**.

Table 2. Size factors S_{ij} (r, retentate; p, permeate).

Stage (j)		**S_{ij}=m^3/kg**			
	Unit	Insulin	Vaccine	Chymosin	Protease
1	Fermenter	1.25	0.625	0.415	0.3125
2	Microfilter I	r: 1.25	r: 0.625	r: 0.415	r: 0.3125
		p: 2.5	p:no	p: 0.830	p: no
3	Homogenizer	No	0.155	No	0.08
4	Microfilter II	No	r: 0.155	No	r: 0.08
			p: 0.31		p: 0.16
5	Ultrafilter I	2.5	0.31	0.83	0.16
6	Extractor	0.4	0.2	0.14	0.1
7	Ultrafilter II	0.4	0.2	0.14	0.1
8	Chromatographer	0.05	0.05	0.05	0.05

Table 3. Time factors T_{ij} [B_i(kg)].

Stage	**Unit**	**T_{ij} (h)**			
j		**Insulin**	**Vaccine**	**Chymosine**	**Protease**
1	Fermentor	24	24	24	24
2	Microfilter I	12.5 $A^{-1}B_i$	2.5 $A^{-1}B_i$	4.15 $A^{-1}B_i$	1.25 $A^{-1}B_i$
3	Homogeneizer	no	0.465 $cap^{-1}B_i$	no	0.24 $cap^{-1}B_i$
4	Microfilter II	no	3.1 $A^{-1}B_i$	no	1.6 $A^{-1}B_i$
5	Ultrafilter I	105$A^{-1}B_i$	5.5 $A^{-1}B_i$	35 $A^{-1}B_i$	3 $A^{-1}B_i$
6	Extractor	1.5	1.5	1.5	1.5
7	Ultrafilter II	18$A^{-1}B_i$	8 $A^{-1}B_i$	4.75 $A^{-1}B_i$	3 $A^{-1}B_i$
8	Chromatographer	0.5	0.5	0.5	0.5

Most of the separation processes information are taken from Asenjo and Patrick [16], the posynomial modeling approach is taken from Salomone and Iribarren [17]. The general batch process literature [18], describes batch stages through a sizing equation and a cycle time that are applied for a product as follows:

$$V_j \geq S_{ij} B_i \quad (1)$$

where V_j is the size of stage j, e.g., m^3 of the vessel, B_i is the batch size for product i , e.g., kg of product exiting from the last stage, S_{ij} is the size factor of stage j product i, i.e., the size needed at stage j to produce 1 kg of final product i and Tij is the time required to process a batch of product i in stage j considering the fermentor and the insulin product as an example. If we estimate a final concentration of 50 kg dry biomass/m^3 that 0.4 of this biomass is proteins and 0.05 of these proteins is insulin, and an overall yield estimate of the process of 0.8 (0.8 of the insulin produced in the fermenter exits the chromatographic column), then the size factor for the fermenter for producing insulin can be estimate as

$$S_{ij} = \frac{m^3}{50kg \times 0.4 \times 0.05 \times 0.8} = 1.25 m^3 \quad (2)$$

Similarly, vaccine, chymosine and cryophilic protease were estimated to be 0.1, 0.15 and 0.2 of total proteins of the biomass, respectively. The batch stage description is completed by estimating a processing time T_{ij} for stage j when producing product i. For the fermenter, we estimate T_{ij}=24 h for all products, which includes time for charging, cell growth, and discharging.

This model of batch stages given by constraint (1a) is the simplest one. Its level of detail suffices for the fermenter and the extractor. These units are truly batch items chat hold the load to be processed and whose operations are governed by kinetics and hence, the operating time does not depend on the batch size. The first approximation for the extractor, we take a phase ratio of (1b) for all products. Therefore, the required extractor volume is twice the inlet batch volume, while the inlet and outlet aqueous saline batches are of the same volume. It is also assumed, as a result of preliminary balances, that this operation reduces the total amount of proteins to about twice the amount of the target protein. with respect to the kinetic effects we take as first estimates [19] the following times: 15 min stirring to approach phase equilibrium, 30 min settling to get almost complete disengaging of the phases and 20 min for charging and discharging. A special consideration must be done in the case of the microfiltration, homogenization and ultrafiltration stages. Although the mathematical model considers them batch stages, their corresponding equipment consists of holding vessels and semicontinous units that operate on the material that is recirculated into the holding vessel. The batch items are sized as described before. For example, for the homogenizer processing cryophilic protease, we estimated that the fermentor broth is concentrated 4 times up to 200 kg/m^3 at microfilter 1 and considered a yield of 1 because the intracellular protease is fully retained at the microfilter. Then the size factor of the homogenizer vessel is 4 times smaller than the fermenters, i.e. S_{ij}=0.08 m^3/kg protease. The sizing equation for semicontinuous items can also be found in the general batch processes literature [20]:

$$R_j = D_{ij} \frac{B_i}{\theta_{ij}} \quad (3)$$

where R_j is the size of the semicontinuous item k, usually a rate of processing. For example, in the case of the homogenizer, it is the capacity in cubic meters of suspension per hour, but in the case of the filters R_j is their area of filtration A_j (m^3). B_i is again the batch size, θ_{ij} is the operating time that the semicontinuous item j needs to process a batch of product i and D_{ij} is the duty factor (a size factor for semicontinuous items), i.e., the size needed at stage j to process 1 kg of product i in 1 h. For example, if we adopt three passes through the homogenizer, its duty factor is the vessel size factor 0.08 m^3/kg×*3*, i.e., D_{ij}=0.24 m^3/kg. The meaning of a capacity of 0.24 m^3/h is that it allows 1 kg of final product cryophilic protease to be processed in 1 h.

The general batch processes literature considers semicontinuous units to work in series with batch units so that their operating time are the times for filling or emptying the batch units. However, in the process considered, pumps are the only semicontinuous units, which transfer batches between the units. As the pumps cost does not have a relevant impact on the plant design, they were not explicitly modeled. The times for filling and emptying batch items were estimated and included in the batch cycle times. On the other hand, the process does have special semicontinuous units with an important economic impact on the cost. They are the homogenizer and ultrafilters, but their operating time is the batch processing time of the respective stage. The mathematical model depends on both the batch size and the size of the semicontinuous item are as follows:

$$V_j \geq S_{ij} B_i \quad (4a)$$

$$T_{ij} = T_{ij}^0 + T_{ij}^1 \frac{B_i}{R_j} \quad (4b)$$

where R_j refers to the size of the semicontinuous item that operates on the batch size at stage j. T^0_{ij} and T^1_{ij} are appropriate time factors that take into account contributions to the total cycle time of the stage that are either fixed amounts of time or proportional to the batch size and inversely proportional to the size of the semicontinuous item. For the homogenizer, R_j is its capacity, T^1_{ij} the duty factor of the homogenizer itself and T^0_{ij} includes the estimated times for filling and emptying the homogenizer holding vessel. In the case of ultrafilters, a fixed permeate

flux model was considered with a rate of 20 L/m^2 of membrane area/h. In this case, the size of the semicontinuous item R_j is the filtration area. T^0_{ij} is again the time for filling and emptying the retentate holding vessel, and T^1_{ij} is the inverse of the permeate flux times the ratio (m^3 permeate/kg). This ratio is estimated from a mass balance taking into account that the ultrafilters are used for a water removal from solutions up to 50 g/L of total proteins. Ultrafilters are used to reduce the volume required at the liquid extractor and the chromatographic column. The upper bound on concentration is a constraint that avoids protein precipitation. The microfilter model is quite similar to that of the ultrafilter, but there are two batch items associated to them instead of one, the retentate and the permeate vessels, plus the semicontinuous item area of filtration. For microfilter 1 a fixed permeates flux of 200 L/m^{2}h is adopted. For extracellular insulin and chymosin, we estimate a total permeate (feedwater plus make up water) twice the feed, while for intracellular protease and vaccine we estimate it in 75% of the feed (the retentate is concentrated four times). For microfilter 2 a fixed permeate flux model is also used. In this case, the flux is smaller than the one in microfilter 1 because the pore size to retain cell debris is smaller than the one for whole cells. As a first estimation we take 100 L/m^{2}h and a total permeate (feed plus make up water) twice the feed. With respect to the chromatographic column, an adsorptive type chromatography is considered, with a binding capacity of 20 kg/m^3 of column packing. The size factor of this unit is the inverse of that binding capacity. As a first approximation, a fixed total operating time of 0.5 h was estimated for loading, eluting and washing regeneration.

Finally, the stage model is completed with a cost model that expresses the cost of each unit as a function of its size, in the form of a power law. These expressions are summarized in **Table 4**, with most of the cost data [20].

Table 4. Cost of equipment (U.S. Dollars).

Unit	Size	Cost
Fermenter	$V_j(m^3)$	$63400V^{0.6}$
Micro and ultrafilters	$V_{retentate}(m^3)$	$5750V_r^{0.6}$
Homogenizer	$)V_{holding}(m^3)$	$5750V^{0.6}$
	$\mathrm{Cap}(m^3/h)$	$12100cap^{0.75}$
Extractor	$V_{extr}(m^3)$	$23100V^{0.65}$
Chromatography	$V_{chrom}(m^3)$	$360000V^{0.995}$

Model equations

The mathematical optimization model for designing the multiproduct batch plant is described in this section. The model includes the stage models described in the previous section plus additional constraints that are explained in this section. The plant consists of M batch stages (in our case 8 batch stages). Each stage j has a size V_j(m^3) and more than one unit can be installed in parallel. They can work either in-phase (starting operation simultaneously) or out of phase (starting times are distributed equally spaced between them). The duplication in phase is adopted in case the required stage size exceeds the specific upper bound. In this case G_j units are selected, splitting the incoming batch into G_j smaller batches, which are processed simultaneously by the G_j units. After processing, the batches are added again into a unique outgoing batch. Otherwise, duplication out-of-phase is used for time-limiting stages, if a stage has the largest processing time, then it is a bottleneck for the production rate. Assigning M_j units at this stage, working in out of phase mode, reduces the limiting processing time and thus increases the production rate of the train. For this case, the batches coming from the upstream stages are not split. Instead, successive batches produced by the upstream stage are received by different units of stage j, which in turn pass them at equally spaced times onto the downstream batch stage. The allocation and sizing of intermediate storage has been included in the model to get a more efficient plant design. The goal is to increase unit utilization. The insertion of a storage tank decouples the process into two subprocesses: one upstream from the tank, and the other downstream. This allows the adoption of independent batch sizes and limiting cycle times for each subprocess.

Therefore, the previously unique B_i is changed to batch sizes B_{ij} defined for product i in stage j. Appropriate constraints adjust the batch sizes among different units. The objective is to minimize the capital cost of the plant. The decision variables in the model are as follows:

At each batch stage the number of parallel units in phase and out of phase and their size, and the installation or absence of intermediate storage between the batch stages and their size. The plant is designed to satisfy a demand of Q_i (kg) of each product i, for the P product considered, within a time horizon $H(h)$.

In summary, the objective function to be optimized is

$$MinCost = \sum_{j=1}^{M} M_j G_j a_j V_j^{\alpha_j} + \sum_{j=1}^{M} VT_j^{\eta_j} \quad (5)$$

where a_j and α_j, c_j and ηj are appropriate cost coefficients that depend on the type of equipment being considered. VT_j is the size of the storage tank allocated after stage j. The size of each unit has to be large enough to be able to process every product:

$$V_j \geq \frac{S_{ij} B_{ij}}{G_j} \forall i = 1,\ldots,P; \forall j = 1,\ldots,M \quad (6)$$

where S_{ij} is the size factor for product i in stage j. In case of parallel units working in phase, the division of B_{ij} by the number of units G_j takes into account the reduction in the batch size to be processed by these units. The operation time T_{ij} to process product i at stage j has the general following form:

$$T_{ij} = T_{ij}^0 + T_{ij}^1 \frac{B_{ij}}{R_j} \quad \forall i = 1,\ldots,P; \forall j = 1,\ldots,M \quad (7)$$

where T_{ij}^0 and T_{ij}^1 are appropriate constants that depend on both the product and the stage. Expression 7 accounts for a fixed and variable contribution to the total operating time. The last term in Equation 7 depends on both the batch size and the size of the semicontinuous item associated to this batch stage, as was already discussed previousely.

The limiting cycle time for product i in the subprocess h, TL^h, is the largest processing time in this production train:

$$TL_i^h \geq \frac{T_{ij}}{M_j} \quad \forall i = 1,\ldots,P; \forall j \in J_j; \forall h \quad (8)$$

where J_h is the set of units which conform the subprocess h the division by the number of units in parallel working out of phase, M_j takes into account the reduction in the cycle time of this stage due to the operation of M_j units that alternatively process the consecutive batches. To avoid accumulation of material, the processing rate of both subprocess downstream and upstream of the storage tank must be the same:

$$\left(\frac{B_i^d}{TL_i^d}\right) = \left(\frac{B_i^u}{TL_i^u}\right) \quad \forall i = 1,2,\ldots,P \quad (9)$$

The constraints 9 equalizes the production rate upstream and downstream of the storage tank. To express 9 in a simple form, the inverse of the production rate of product $i(E_i)$, is defined as:

$$E_i = \frac{TL_i^h}{B_{ij}} \quad \forall i = 1,2,\ldots,P; \forall j \in J_h; \forall h \quad (10)$$

Expression 10 is used to replace TL_i^n in constraint 8, dropping constraint 9. The production constraint is posed as follows: during the time horizon H the plant must produce the target production quantities Q_i of each product i. The number of batches of each product i to be produced during time H is $\frac{Q_i}{B_i}$ and the production of each batch demands a time TL_i. The following constraints holds:

$$\sum_{i=1}^{P} Q_i E_i \leq H \quad (11)$$

The size of the storage tank VT_j, allocated after batch stage j, is given by the following expression [25]:

$$VT_j \geq ST_{ij}\left(B_{ij} + B_{ij+1}\right) \forall i = 1,\ldots,M; \forall j = 1,\ldots,M-1 \quad (12)$$

where ST_{ij} is the size factor corresponding to the intermediate storage tank, with identical definition to the batch stages. As no a priori tank allocation is given, binary variables y_j are used to select their allocation. The value of variables y_j is 1 if a tank is placed in position j, or zero otherwise. Constraint 12 is generalized to size the tank only if it exits:

$$VT_j \geq ST_{ij}\left(B_{ij} + B_{ij+1}\right) - F_j\left(1 - y_j\right) \forall i = 1,\ldots,P; \forall j = 1,\ldots,M-1 \quad (13)$$

where F_j is a constant value sufficiently large such that when y_j is 0 (the tank does not exist), the constraint is trivially satisfied for any value of VT_j.

In particular, the cost minimization will drive VT_j=0. When the tank exists (y_j=1) the term with F_j vanishes and the original constraint (12) holds. If the storage tank does not exist between two consecutive stages, then their batch sizes are constrained to be equal. Otherwise, this constraint is relaxed. This effect is imposed by the following constraints:

$$1 + \left(\frac{1}{\Phi} - 1\right) y_j \leq \frac{B_{ij}}{B_{ij+1}} \leq 1 + (\Phi - 1) y_j \quad \forall i = 1,\ldots,P; \forall j = 1,\ldots,M-1 \quad (14)$$

where φ is a constant value corresponding to the maximum ratio allowed between two consecutive batch sizes.

In summary, the multiproduct plant design model that includes the options of parallel units in-phase and/or out of phase and provision of intermediate storage, consists of the objective function 5 subject to constraints 6, 8, 11, 13 and 14, plus the upper and lower bounds that may apply. An important feature of the model is that both the objective function and the constraints are posynomial expressions that possess a unique local (and thus, global) solution [20]. This basic model has been adapted to handle the particular feature of the composite stages (homogenizer, ultrafilters and microfilters). In this case, constraint 6 is applied not to a

general batch stage size but to each of the items that compose it. So in the case of microfilters, constraint 6 applies to both the retentate and the permeate vessels. A new parameter SR_{ij} was introduced to represent the size factor of the retentate vessel, while S_{ij} was left for the permeate vessel. Also in this case, the objective function must account for all the stage components. The notation a_j and α_j were left for the cost coefficients of the permeate vessel, b_j and β_j for the retentate vessel and d_j and γ_j for the filtration area. A similar approach was implemented for the ultrafilters (retentate vessel and ultrafiltration area) and homogenizer (holding vessel and the homogenizer itself).

Methodology

Between 2000s and 2020s witnessed a tremendous development in the size and complexity of industrial organizations. Administrative decision-making has become very complex and involves large numbers of workers, materials and equipment. A decision is a recommendation for the best design or operation in a given system or process engineering, so as to minimize the costs or maximize the gains [21]. Using the term "best" implies that there is a choice or set of alternative strategies of action to make decisions. The term optimal is usually used to denote the maximum or minimum of the objective function and the overall process of maximizing or minimizing is called optimization. The optimization problems are not only in the design of industrial systems and services, but also apply in the manufacturing and operation of these systems once they are designed. Including various methods of optimization, we can mention: MINLP, route-packing-based-batch (RPBB) and seed-based-batch (SeedBB).

Route-packing-based batching (RPBB)

Binning packs items into bins so that the structure of each bin aids the formation of batches with short total travel distances. The objective is to shorten the total travel distances of all bins since it is easier to form batches with short total travel distances from the bins with short total travel distances formulating the route-selection-based binning model to group items into bins and select a route for each batch. The heuristic repeats until no more items remain to be batched as follow [22]:

- Step 1. Sort all items in descending order of size.
- Step 2. If there are no more items on the list, terminate; otherwise, select the first item inthe list, place it into a new batch as the current batch and delete the item from the list.
- Step 3. If it is the last item on the list, set the current bin as complete and go back to step 2; otherwise, move to the next item on the list and set this item as the candidate item for the current bin.

Step 4. If the total size of current bin and candidate item is smaller than or equal to the bin capacity, add the candidate item to the bin, delete the candidate item from the item list and go back to step 3.

Seed-based-batch (SeedBB)

The seed heuristic constructs a batch starting from the selection of seed order as the batch's initial order and adds orders iteratively until the batch is full or no orders remain to be batched. Seed-based binning-then-batching (SeedBB) uses seed heuristics to solve the binning problem and the batching problem separately as follows [23]:

Step 1. Calculate the saving in travel distance S_{ij} for all possible bin pairs i,j considering the size of each bin and batch capacity.

Step 2. Select the pair with the highest saving as the best pair. In case of a tie, select a random pair.

Step 3. Combine the bins in the best pair as a single bin and update the bin list.

Step 4. If there is any bin with size smaller than or equal to the batch capacity, go back to step 1; otherwise, put each of the remaining bins into an individual batch and terminate.

Statistical analysis methods

The interest in statistical analysis methods has grown recently in the field of computational intelligence. In this section, I will discuss the basic and give a survey of a complete set of variance analysis procedures developed to perform the comparison between RPBB and SeedBB, via the use of describing a test of the null hypothesis, which applies to independent random samples from two normal populations of size n_1 and n_2 are taken from normal population having the same variance, it follows distribution with n_1-1 and n_2-1 degrees of freedom, according to this equation: —

However, the error from the optimal solution is given by:

$$error\% = 100\frac{\left|x_{\exp} - x_{cal}\right|}{x_{\exp}} \quad (19)$$

In this research, x_{exp} is considered to be the optimal solution founded by Montagna (Plant cost $829,500), where the equation (19) is a criterion to confirm the optimal values.

RESULTS

The problem could be formulated as the minimization of the investment cost for equipment and storage tanks. Given that the problem modeled has nonlinear objective function. For the purpose of optimization problem, the model developed has been solved with RPBB and SeedBB Python Toolbox respectively, which is included in the GNU Octave Scientific Programming Language, using the data shown in **Tables 1, 2, 3 and 4**. A horizon time of 6000 h has been considered.

However, the intermediate storage cost coefficient with size factors is shown in **Table 5.**

Table 5. Intermediate storage cost coefficients and size factors.

Unit[a]	***ST_{ij}* size factor for product i in stage j**			
	Insulin	**Vaccine**	**Chymosin**	**Protease**
Fermenter	1.2557	0.6254	0.4151	0.3128
Microfilter I	2.5013	0.1557	0.8301	0.0805
Homogenizer	2.5014	0.1553	0.8307	0.0809
Microfilter II	2.5047	0.3108	0.8307	0.1608
Ultrafilter I	0.4071	0.2083	0.1357	0.1077
Extractor	0.4012	0.2008	0.1357	0.1007
Ultrafilter II	0.0556	0.0557	0.0517	0.0545
Chromatography	0.0001	0.0003	0.0008	0.0004

On the other hand, the **Table 6** shows the comparison of results for 30 runs between RPBB and SeedBB.

Table 6. Comparison of results for 30 runs between RPBB and SeedBB.

Values	**RPBB ($)**	**SeedBB ($)**
Best	912,4776	833,1208
Average	948,0710	850,319.1979
Worst	976,321.1999	865,492.2013
Standard deviation	9.2019	1.2006

Nevertheless, the optimization runs result for the investment cost calculated by RPBB and SeedBB during 30 runs is illustrated in **Table 7.**

Table 7. Optimization runs results for the investment cost founded by RPBB and SeedBB during 30 runs.

Technique	**Plant cost ($)**	**% from optimal solución**	**CPU time (s)**
RPBB	912,4776	0.5	100
SeedBB	833,1208	11	900

Nonetheless, the equipment structure computed by SeedBB is showed in **Table 8.**

Table 8. Equipment structure calculated by SeedBB.

Stage	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
V_j	24.7456	———	1.1814	———	9.922	0.8921	0.6017	0.0825
R_j	NA	A: 16.2041	Cap: 1.0989	A:8.668	A: 109.3301	NA	A: 17.8134	NA
VT_j	29.7066	NA	NA	NA	2.2154	NA	0.3795	NA
M_j	3	3	3	3	3	3	3	3
G_j	3	3	3	3	3	3	3	3

However, **Table 9** shows equipment structure calculated by RPBB.

Table 9. Equipment structure calculated by RPBB.

Stage	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
	22.6085	———	1.0794	———	9.0651	0.8151	0.5497	0.0754
	NA	A: 14.8047	Cap: 1.004	A: 7.9194	A: 99.8880	NA	A: 16.2750	NA
	27.1410	NA	NA	NA	2.0241	NA	0.3795	NA
	1	1	1	1	1	1	1	1
	1	1	1	1	1	1	1	1

The idle times in plant calculated by SeedBB is provided in **Table 10**.

Table 10. Idle times in plant calculated by SeedBB (s).

Unit								
Product	1	2	3	4	5	6	7	8
Insulin	0	0	NA	NA	0	57.7	NA	67.11
Vaccine	0	54	0	0	60.7977	57.7128	22.14	67.13
Chymosin	0	17	NA	NA	17.5501	57.7708	27.9171	67.1178
Protease	0	63	16	15	63.0713	57.7794	55.0317	67.1103

However, the idle times in plant calculated by RPBB is shown in **Table 11**.

Table 11. Idle times in plant calculated by RPBB (s).

Unit								
Product	1	2	3	4	5	6	7	8
Insulin	0	0	NA	NA	0	0.01	0	0
Vaccine	0	1.91	0.03	0	2.81	0	0.15	0
Chymosin	0	0.01	NA	NA	0	0	0.31	0.16
Protease	0	2.05	0	0	3.05	0	0.3	0

The results of the statistical analysis are illustrated in **Tables 12** and **13**.

Table 12. The results of two algorithms solving MBPD problem.

Algorithm	**N**	**Avg**	**SD**	**Standard Error**	**95% Confidence Interval of Mean**		**Min**	**Max**
					Min	**Max**		
SeedBB	30	1879.0000	8.68935	2.98743	1933.9205	1945.0795	1928	1957
RPBB	30	1831.0000	5.39936	2.05701	1728.2733	1737.7201	1728	1745

Table 13. Variance analysis result of MBPD problem.

	Quadratic sum	**Free degree**	**Mean square**	**F**	**Significance**
SDB	2349.676	3	778.895	14.455	0.000
SDI	1824.100	36	51.392	-	-
SUM	4354.775	39	-	-	-

DISCUSSION

It is clear from the summary of the results shown in **Table 7**, that the performance of both SeedBB and RPBB produce adequate values regarding the cost for equipment and storage tanks. However, RPBB performs better than the SeedBB in terms of the average and the worst fitness values and the standard deviation. **Table 7**, also, shows the best final solution found in the 30 runs of SeedBB and RPBB. According to our knowledge, the case study about the optimal design of protein production plant has been taken from Montagna. However, they solved the problem using rigorous mathematical programing (MINLP), their model includes 104 binary variables and has been convexified using the transformation proposed by Kocis and Grossmann. The MINLP model has been solved using DICOPT++, which is included in the GAMS optimization modeling software. The algorithm implemented in DICOPT++ relies on the Outer Approximation/Equality Relaxation/Augmented Penalty (OA/ER/AP) method. The OA/ER/AP solution method consists of the decomposition of the original MINLP problems into a sequence of two subproblems: A nonlinear programming (NLP) subproblem and a mixed integer linear programming (MILP) subproblem also known as the Master problem, which is solved to global optimality (minimize the caplital cost $829,500). However, in previous work of Montagna and other, their model needed a long computational time (more than 86400 s) and require several initial values to the optimization variables, they also showed in their paper that the behavior of the demand was completely deterministic.

Whilst, this assumption does not seem to be always a reliable representation of the reality, since in practice the demand of pharmaceutical products resulting from the batch industry is usually variable.

Simulations outcomes were then compared with experimental data in order to check the accuracy of the method. **Table 7** presents the results obtained in different optimization runs for multiproduct batch plant design. For each simulator run, the average numerical effort spent on solving the problem on LINUX System, Intel® D, CPU2.80 GHz, 2.99 of RAM. **Table 7** shows plant cost, % from optimal solution and CPU time obtaining during 30 runs. SeedBB and RPBB performed effectively and give a solution within 10 and 0.5% of the global optimal $912,450 and $833,647, respectively. Furthermore, the important feedback could be taken from **Table 7**, is the GA results in a faster convergence than SeedBB and the MINLP algorithm. In addition, the RPBB is so close to the global optimal of MBPD (0.5% from optimal solution) and provides also an interesting solution, in terms of quality as well as of computational time as illustrated in **Table 7**, while **Table 8** presents the sizes for the units involving a set of discrete equipment structure given by SeedBB. The inconvenience of this configuration is just stopped at 6000 h with risk of failing to fulfill the potential future demand coming from a fluctuation of the market.

In order to show how the evolution process is going on for both SeedBB and RPBB, respectively, the convergence of the best fitness values. The convergence rate of objective function values as a function of generations for both SeedBB and RPBB where for clarity only 1000 generations are shown. For the optimization problem considered, RPBB decrease rapidly and converge at a faster rate (around 500 generations) compared to that for SeedBB (about 800 generations), from which it is clear that RPBB seem to perform better compared to SeedBB. So, for the present problem the performance of the RPBB is better than SeedBB from an evolutionary point of view.

To compare the computational time, the swarm/population size is fixed to 200 for both SeedBB and RPBB algorithms. Whereas, the generation number is varied. Simulation were carried out and conducted on LINUX System, Intel (R) D, CPU 2.80 GHz, 2.99 of RAM Computer, in the GNU Octave environment. Here the result in the form of graph is shown in. It is clear from that the computational time for RPBB is very low compared to the SeedBB optimization algorithm. Further, it can also be observed from hat in case of RPBB the computational time increases linearly with the number of generations, whereas for SeedBB the computational time increases almost exponentially with the number of generations. The higher computational time for SeedBB is due to the communication between the particles after each generation. Hence as the number of generations increases, the computational time increases almost exponentially.

Table 8 presents the sizes for the units involving a set of discrete equipment structure given by SeedBB. The inconvenience of this configuration is just stopped at 6000 h with risk of failing to fulfill the potential future demand coming from a fluctuation changing of the market.

On the other hand, the calculation of the structure of equipment using RPBB is illustrated in **Table 9**. The total production time, also, computed by RPBB is 5491.12 h to fulfill the eventual increase of future demand caused by market fluctuation. In addition, the RPBB results in a faster convergence. However, the equipment structure showed by SeedBB is very expensive. Furthermore, the SeedBB approach has the disadvantage of long CPU time.

At the same time as, the RPBB allow the reduction of the idle time to the stage, in any way, **Table 10** and **Table 11** show the idle times obtained by SeedBB and RPBB respectively.

However, some observations about some important aspects in our implication of RPBB and some problems in practice: The most important of all is the method of coding, because the codification is very important issue when a Route-packing-based batching is designed to dealing with combinatorial problem, also of the characteristics and inner structure of the DMBP.

The commonly adopter concatenated, multi-parameter, mapped, fixed point coding are not effective in searching for the global optimum. According to the inner structure of the design problem of multiproduct batch that gives us some clues for designing the above mixed continuous discrete coding method with a four-point crossover operator. As is evident from the results of application, this coding method is well fit for the proposed problem.

In order to further explain the effects of these algorithms on solving the MBPD problem, the variance analysis was performed. Each of the SeedBB and RPBB algorithms was run 30 times. The R software was used to analyze the results. Therefore, the results are given in **Table 12** and **13**.

Table 13 indicates that, the mean square deviation between groups (SDB) is 779.895. The mean square deviation within groups (SDI) is 50.392. The test statistic F = 15.477. If significance level $\alpha = 0.05$, then the critical value $2.92 \leq F_{\alpha}(3.36) \leq 2.84$. Thus, $F > F_{\alpha}(3.36)$ indicating that the difference between the average is significant, that is, the performance difference of algorithms is significant.

Nevertheless, these techniques are not a panacea, despite their apparent robustness, there are control “parameters” involved in these metaheuristics and appropriate setting of these parameters is a key point for success.

CONCLUSION

Techniques such as SeedBB and RPBB are inspired by heuristic mechanism nature and have proved themselves to

be effective solutions to optimization problems. We applied route-packing-based batching to solve the problem of DMBP. RPBB perform effectively and give a solution within 0.5% of the global optimum. Whilst, it is observed that, in terms of computational time, the RPBB approach is faster. The computational time increases linearly with the number of generations for RPBB, whereas for SeedBB the computational time increases almost exponentially with the number of generations, interpreting that, the higher computational time for SeedBB is due to the communication between the particles after each generation. Furthermore, the results provided by RPBB are much better with respect to SeedBB. In this paper, the RPBB gave us the highest efficiency and justifies its use for solving nonlinear mathematical models. Therefore, this work provides an interesting decision/making approach to improve the design of multiproduct batch plants under conflicting goals.

REFERENCES

1. Dietz A, Azzaro-Pantel C, Pibouleau L, Domenech S (2006) Multiobjective optimization for multiproduct batch plant design under economic and environmental considerations. Comput Chem Eng 30: 599-613.
2. Dietz A, Azzaro-Pantel C, Pibouleau L, Domenech S (2008) Strategies for multiobjective genetic algorithm development: Application to optimal batch plant design in process systems engineering. Comput Ind Eng 54: 539-569.
3. Ponsich A, Azzaro-Pantel C, Domenech S, Pibouleau L (2007) Mixed-integer nonlinear programming optimization strategies for batch plant design problems. Ind Eng Chem Res 46: 854-863.
4. Dietz A, Aguilar-Lasserre A, Azzaro-Pantel C, Pibouleau L, Domenech S (2008) A fuzzy multiobjective algorithm for multiproduct batch plant: Application to protein production. Comput Chem Eng 32: 292-306.
5. Aguilar-Lasserre AA, Bautista MA, Ponsich A, González Huerta MA (2009) An AHP-based decision-making tool for the solution of multiproduct batch plant design problema under imprecise demand. Comput Oper Res 36: 711-736.
6. Aguilar-Lasserre A, Giner AH, Azzaro-Pantel C, Guillermo CR, Constantino MS, et al. (2009) Multiobjective multiproduct batch plant design under uncertainty: Application to protein production. 10th International Symposium on Process Systems Engineering - PSE2009, August 15-20, 2009, Salvador, Brazil.
7. Borisenko A, Kegel P, Gorlatch S (2011) Optimal design of multi-product batch plants using a parallel branch-and-bound method. Springer, Berlin.
8. Hasebe S (1998) Optimal scheduling and minimum storage tank capacities in a process system with parallel batch units. Comput Chem Eng 3: 185-195.
9. Cao DM, Yuan XG (2002) Optimal design of batch plants with uncertain demands considering switch over of operating modes of parallel units. Ind Eng Chem Res 41: 4616-4625.
10. Crougham M, Caldwell V, Randlev B, Billeci K, Nieder M (1997) Prediction of culture performance through cell cycle analysis: A potential tool in operations scheduling. The Biochemical Engineering X Conference, Kananaskis, Canada.
11. Montagna JM, Iribarren OA, Galiano FC (1994) The design of multiproduct batch plans with process performance models. Chem Eng Res Des 72: 783.
12. Azzaro-Pantel C, Dietz A, Pibouleau L, Domenech S (2005) A framework for multiproduct batch plant design with environmental consideration: Application to protein production. Ind Eng Chem Res 44: 2191-2206.
13. Dessouky Y, Bayer A (2002) A simulation and design of experiment modeling approach to analyze facility maintenance costs. Comput Ind Eng 43: 423-436.
14. Montagna JM, Vecchietti AR, Iribarren OA, Pinto JM, Asenjo JA (2000) Optimal design of protein production plants with time and size factor process models. Biotechnol Prog 16: 228-237.
15. Birattari M, Zlochin M, Dorigo M (2006) Towards a theory of practice in metaheuristics design: A machine learning perspective. Theor Info Appl 40: 353-369.
16. Petrides D, Sapidou E, Calandranis J (1995) Computer aided process analysis and economic evaluation. Biotechnol Bioeng 48: 529-541.
17. Asenjo JA, Patrick I (1990) Protein Purification Applications: A Practical Approach. IRL Press: Oxford Chapter I, pp: 1-28.
18. Salomone HE, Iribarren OA (1992) Posynomial modeling of batch plants. Comput Chem Eng 16: 173-184.
19. Mistry SL, Kaul A, Merchuk JC, Asenjo JA (1996) Mathematical modeling and computer simulation of aqueous two-phase continuous protein extraction. J Chromatogr A 741: 151-163.
20. Modi AK, Karimi IA (1989) Design of multiproduct batch processes with finite intermediate storage. Comput Chem Eng 13: 127-139.
21. Grossmann IE, Sargent RW (1979) Optimal design of multipurpose chemical plants. Ind Eng Chem Process 18: 343-348.
22. Hong S, Kim Y (2017) A route-selecting order batching model with the S-shape routes in a parallel-aisle order picking system. Eur J Oper Res 257: 185-196.
23. Hong S, Johnson AL, Peters BA (2012) Large-scale order batching in parallel-aisle picking systems. IIE Trans 44: 88-106.