18th European Symposium on Computer Aided Process Engineering – ESCAPE 18
Bertrand Braunschweig and Xavier Joulia (Editors)

Product Portfolio Management with Discrete-Event Simulation and Genetic Algorithm

J.L. Pérez-Escobedo[a], A. Aguilar-Lasserre[b], C. Azzaro-Pantel[a], L. Pibouleau[a], S. Domenech[a]

[a]*Laboratoire de Génie Chimique de Toulouse, UMR 5503 CNRS, ENSIACET INPT*
5 rue Paulin Talabot, BP1301 – 31106 Toulouse Cedex – France
[b]*Instituto Tecnologico de Orizaba-Av. Oriente 9, n. 852 (A.P. 324), Col. Emiliano Zapata 94320, Orizaba, Veracruz*

Abstract

This paper presents a discrete-event simulation environment that models the underlying structure of a pharmaceutical enterprise portfolio. An object-oriented model structure previously developed for batch plant scheduling and design is extended to embed the case of product management, which is particularly adequate for re-use of both structure and logic. By wrapping such a simulation with advanced optimization approaches, such as Genetic Algorithms, it becomes possible to evaluate a large set of scenarios for the pharmaceutical enterprise, with a holistic fashion that avoids local optima that can be detrimental to the enterprise.

Keywords: product management, discrete-event simulation, optimization, Genetic Algorithms.

1. Introduction

A fundamental challenge in managing a pharmaceutical or biotechnology company is identifying the optimal allocation of finite resources across the infinite constellation of available investment opportunities. In that context, the optimal management of the new product pipeline has emerged at the forefront of all strategic planning initiatives of a company.

This issue is traditionally identified as a complex one since it integrates various areas such as product development, manufacturing, accounting and marketing. The complexity of the problem is mainly attributed to the great variety of parameters and decision-making levels involved. A strategic investment plan should simultaneously address and evaluate in a proper manner the following four main issues: product management, clinical trials uncertainty, capacity management and trading structure. It is also generally viewed as a multistage stochastic portfolio optimization problem. The main challenge is to configure a product portfolio in order to obtain the highest possible profit, including any capacity investments, in a rapid and reliable way. These decisions have to be taken in the face of considerable uncertainty as demands, sales prices and outcomes of clinical tests that may not turn out as expected.

This kind of problem has recently received attention from the process systems engineering community utilizing previous works from the process planning and scheduling area. Schmidt and Grossmann (1996) proposed various MILP optimization models for the scheduling of testing tasks with no resource constraints with a discretization scheme in order to induce linearity in the cost of testing. Jain and Grossmann (1999) extended these models to account for resource constraints.

Subramanian et al. (2003) proposed a simulation-optimization framework that takes into account uncertainty in duration, cost and resource requirements and extended this model to account for risk. Maravelias and Grossmann (2001) proposed an MILP model that integrates the scheduling of tests with the design and production planning decisions. A literature review of optimization approaches in the supply chain of pharmaceutical industries can be found in Shah (2003). A recent paper was presented by Blau et al. (2004) with a monoobjective Genetic Algorithm to optimize product sequence evaluated by a commercial discrete-event simulator.
This paper lies in this perspective : the underlying idea is to use a multiobjective fuzzy framework as already initiated by (Aguilar-Lasserre et al., 2007) to model both the conflicting nature of the criteria (i.e. risk minimization and profitability maximization) and the imprecise nature of some parameters (demand, operating times …). In that context, this work aims at the development of an architecture that combines an optimization procedure based on a multiobjective genetic algorithm and a discrete-event system (DES) simulation to assess the uncertainty present in the pipeline and to help decision-making. The presentation is mainly focused on the development of the simulator initially developed for scheduling, planning and design purposes.

2. Problem formulation

Within the scope of Research and Development Pipeline management problem, several new-product-development (NPD) projects compete for a limited pool of various resource types. Each project product usually involves a series of testing tasks prior to product commercialization. If the project fails any of these tasks, then all the remaining work on that product is stopped and the investment in the previous testing tasks is wasted. In its most general form, the R&D Pipeline management problem can be formulated as follows: given a set of products that are potential candidates and a set of resources to complete the testing tasks. Each potential product is required to pass a series of tests. Each test has a given duration, cost and probability of success that is assumed to be known a priori. Resources can also be installed at a known cost. Resources are discrete in nature (e.g. laboratories, scientists) and each resource unit can handle only one task at a time. Tests are assumed to be non-preemptive. A flow diagram of the activities involved in the development of a new pharmaceutical product is proposed in Fig. 1. Although some differences may exist referring to various industrial practices, we consider it as generic enough to embed various formulations.

The idea is to model the various paths and the precedence relations between these activities by discrete-event simulation principles used in previous works for batch plant design. The problem of evaluating and selecting which new products to develop and then of sequencing or of scheduling them is not a trivial task due to dependencies between products both in the market place and in the development process itself.

3. Discrete-event Simulation

3.1. Process Modelling.

In a DES, a process is described as it evolves with time and changes take place only a finite number of times, i.e. event occurrence date. The DES was developed using C++ object-oriented language, keeping the approach proposed by (Bérard et al., 1999) (Fig. 2). The power of object oriented techniques lie in the ability to produce “modular” code (known as classes) that can be easily modified and reused. The four layer framework proposed by (Bérard et al., 1999) {engine, event, object, supervisor} is used again in this work. At the lowest level, the common engine is found. Initially, the events in the next level are generic events common to all batch plant simulations: in this case, the definition must be adapted since we have to consider the whole life cycle of a project related to a product. In the same way, the objects taken into account present some similarities but differ in their appreciation: for instance, in batch plant scheduling problems (BPS), material resources are constituted by equipment whereas in NPD problems, resources may be viewed more globally. In fact, the main differences at this step occur from a terminology point of view and this can be easily transposed in the NPD formulation (see Table 1).

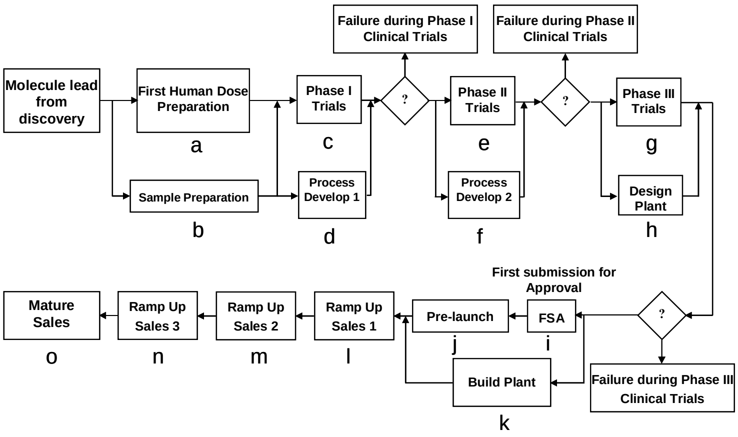

Fig. 1. Flow diagram of the activities involved in the development of a new pharmaceutical product (FHDP: First Human Dose Preparation, FSA, First Submission for Approval)

Table 1. Terminology in BPS and NPD project problems

Batch plant scheduling (BPS)	NPD project
Product #i	Project related to a product #i (PRP #i)
Equipment item #j	Resource of a given step #j
Recipe #k	Succession of activities #k (also called *recipe*)
Unit operation of a recipe #l	Activity #l

Following the classical terminology used in object-oriented approaches, the main so-called objects of the DES are described.

The core of the simulator is the *Engine*, which has two functions: the former is to order the *Events* in its *Calendar* by their *occurrence date* whereas the latter is to activate them if the necessary resources are available; if not, it reports the *Event* to a next *Date*. As previously, an *Event* represents a change of the real system at a given time. The class *Event* is a basis class from which the different *Events* must be defined. An *Event* is characterized by its *occurrence date*, its action over the system and a *type* that enables to give priorities when two or more *Events* have the same *occurrence date*. As a general rule, *Events* which release resources have priority over the others, and when Events have the same type, the classical FIFO rule (First In First Out) is applied. This will be useful when different projects compete for the same resources. The *Event* Class previously developed was generic enough to embed the NPD formulation.

The project for Product # i is a basis class which has the name of the product and its recipe as member data. Each product is dedicated to a disease type. Success probabilities occurring at Phases 1, 2, and 3 (α_1, α_2, α_3), capital cost data (Min, Max) and the level of mature sales (Min, Max) must also be defined. A degree of difficulty (DoD) for computation of capital cost and mature sales is assigned to each product, reflecting the difficulty to carry out the project. A DoD of 1 is assigned to the Min value (respectively 10 for the Max value). The resulting value, either cost or sales, is then computed by linear interpolation from DoD.

The *Recipe* contains the information about the treatment sequence describing the life cycle of a project. In NPD problems, all the projects follow the same *Recipe:* only the duration or cost may differ from a project to another one. This is quite similar to the multiproduct case in BPS. Each *Activity* is described by its name, duration, cost and total available resources. A *Basic Activity* implies that a project step follows a linear path (Simple Input, Simple Output) with a given operating time. Two events are used by the *Activity* Class, respectively *Load* and *Release* classes.

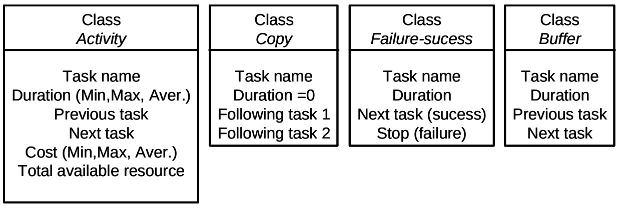

Fig. 2. Some basic classes involved in the DES model.

Additional classes are introduced to model the process described in Fig. 1 for node management. For instance, the *Copy* class takes a project phase of a product as input and distributes the following activities to 2 resources with an operating time equal to 0.

The *failure-success* class is also taken into account to model both the premature stop of the project and its following steps in case of success. Let also note that a so-called *buffer* class is also defined to model that an activity initiated in parallel with another one is waiting for it before passing through the following step. The resulting process structure used in the model is then illustrated in Fig. 3.

Four frequently occurring types of dependencies are also considered in the model (see (Blau et al., 2004) with the same principles: (1) resource dependencies (2) manufacturing cost dependencies; (3) financial return dependencies; and (4) technical success dependencies.

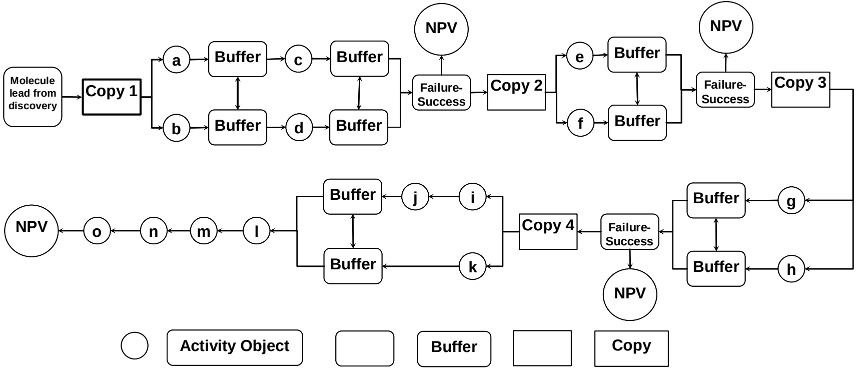

Fig. 3. Network of activities for the DES model.

3.2. Simulation principles

To take into account the stochastic nature of the problem, the simulation is repeated a large number of times selecting random sampling values from the probability distributions and gathering the results, it is possible to determine probability distributions for various economic and risk indicators. The implementation of the simulation thus requires a pre-processing task to generate random numbers for success probabilities. The computation of the Net Present Value (NPV) is cumulated along the evolution of a project for a given product. The so-called reward/risk ratio, obtained by dividing the mean reward by the mean loss, to measure the attractiveness of new product candidates is also evaluated for each project. The mean reward (respectively risk) is calculated from the mean values of positive (respectively negative) NPV values. In that case, the simulation was run 10 000 times.

3.3. Typical results

The detailed presentation of a simulation example will be too exhaustive. Only typical results are analysed. The data set is taken from (Blau et al. 2004) and (Blau, 2000). The problem case study considers nine new drug candidates (D1 to D9) targeted to treat three different diseases. The example inspired from real data is designed to take into account interdependencies between the products. Yet, a direct simple application of the DES model is first to examine the behaviour of each drug candidate individually.

For this purpose, the bubble chart in Fig. 4 provides a graphical view of the project portfolio risk-reward balance. It is used to assure balance in the portfolio of projects-neither too risky or conservative and appropriate levels of reward for the risk involved.

If products with the higher ratio and the higher success probability (upper-right quadrant, here Product 3) constitute the best options, the size of the bubble proportional to the expected capital cost can be another argument.

4. Genetic Algorithm

The second phase of the methodology is to embed the DES in an outer optimization loop. A multiobjective Genetic Algorithm based on the principles developed by (Deb et al., 2002) (Elitist Non-Dominated Sorting Genetic Algorithm, NSGA II) is used.

The focus for the GA is to find a sequence that maximizes the NPV and minimizes the probability of failure taking into account a sequence. A typical Pareto front is presented in Fig. 5 with an interesting compromise sequence.

5. Conclusions and perspectives

This paper has proposed a general framework for the determination of an optimal portfolio for the development of new pharmaceutical products. The approach is based on the combination of a discrete-event simulation model with a multiobjective genetic algorithm procedure. The presentation was mainly focused on the extension of an object-oriented DES model previously developed for scheduling and design of batch processes. The method provides a uniform treatment of both project uncertainties and dependencies, which are inherent to this industry. The use of the mutiobjective AG is particularly adequate to consider the highly combinatorial portfolio management problems facing modern pharmaceutical businesses.

Further works are now devoted to model with more realism the imprecise nature of some parameters, for instance, demand or cost, by fuzzy concepts.

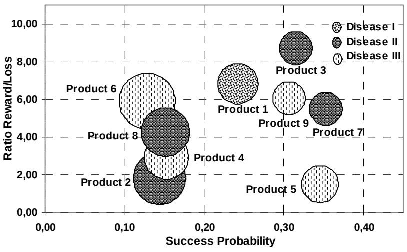

Fig. 4. Results for Reward/Loss ratio for nine products.

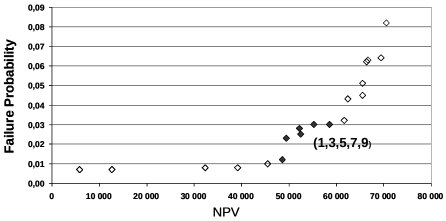

Fig. 5. Typical Pareto Front

References

Aguilar-Lasserre A., , C. Azzaro-Pantel, L. Pibouleau, S. Domenech Enhanced Genetic Algorithm-based Fuzzy Multiobjective Strategy to Multiproduct Batch Plant Design, IFSA World Congress, Cancun, Mexico, June 18-21, 2007, Theory and Applications of Fuzzy Logic and Soft Computing,

Bérard F., C. Azzaro-Pantel, L. Pibouleau, S. Domenech, D. Navarre, M. Pantel, 1999, Towards an incremental development of discrete-event simulators for batch plants : use of object-oriented concepts Escape 9, Budapest, (Hongrie) 31 Mai - 2 Juin, 1999, Comp. And Chem. Eng. Supplements, pp. S565-S568.

Blau G. E., 2000, Risk Management in the development of new products in highly regulated industries, Computer & Chemical Engineering, 24, 659-664.

Blau G. E., J. F. Pekny, V. A. Varma, and P. R. Bunch, 2004, Managing a portfolio of interdependent new product candidates in the pharmaceutical industry, The Journal of Product Innovation Management, 21, 227-245.

Deb K. , A. Pratap, S. Agarwal, and T. Meyarivan, 2002, A fast and elitist multiobjective genetic algorithm: NSGA--II. IEEE Transactions on Evolutionary Computation, 6(2):182—197.

Dietz A., C. Azzaro-Pantel, L. Pibouleau, and S. Domenech, 2005, A Framework for Multiproduct Batch Plant Design with Environmental Consideration: Application To Protein Production, Industrial Engineering and Chemistry Research, 44, pp. 2191-2206.

Jain V., I.E. Grossmann, 1999, Resource-constrained Scheduling of Tests in New Product Development. Ind. Eng Chem. Res,. 38, 3013-3026.

Maravelias, C.; Grossmann, I.E., 2001, Simultaneous Planning for New Product Development and Batch Manufacturing Facilities. Ind. Eng. Chem. Res., 40, 6147-6164.

Shah, N. Pharmaceutical Supply Chains: Key Issues and Strategies for Optimization. In Proceedings of the Fourth International Conference on Foundations of Computer-Aided Process Operations), CACHE, 2003, 73-85.

Schmidt C. W., I.E. Grossmann, 1996, Optimization Models for the Scheduling of Testing Tasks in New Product Development. Ind. Eng. Chem. Res., 35, 3498-3510.

Subramanian D., J.F. Pekny, G.V. Reklaitis, G.E. Blau, 2003, Simulation-Optimization Framework for Stochastic Optimization of R&D Pipeline Management. AIChE Journal, 49(1), 96-112.