PRACTICAL OPTIMIZATION BY OR AND SIMULATION

Nico van Dijk[1], Erik van der Sluis[1]

[1] University of Amsterdam, Faculty of Economics and Business,
Roetersstraat 11, 1018 WB, Amsterdam, the Netherlands

n.m.vandijk@uva.nl (Nico van Dijk)

Abstract

Should we pool capacities or not? This is a question that one can regularly be confronted with in operations and service management. It is a question that necessarily requires a combination of queueing (as OR discipline) and simulation (as evaluative tool) and further steps for '*optimization*'.

It will be illustrated that a combined approach (SimOR) of Simulation (techniques and tools) and classical Operations Research (queueing, linear programming and scheduling) can be most beneficial.

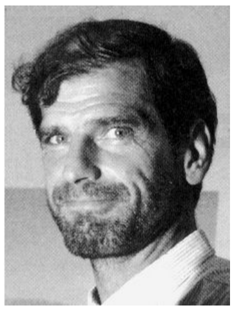
First, an instructive example of parallel queues will be provided which shows the necessary and fruitful combination of queueing and simulation. Next, the combined approach will also be illustrated for the optimization of:

- call centers,
- checking-in at airports,
- blood platelet production.

Whether we should pool or not is thus just one simple question for which this SimOR approach can be most fruitful if not necessary for 'practical optimization'.

Keywords: Optimization, call centers, check-in, blood platelet production, simulation.

Presenting Author's biography



Professor Nico M. van Dijk is responsible for the Operations Research and Management Program at the University of Amsterdam and principal consultant of the simulation company: Incontrol Enterprise Dynamics.

As a scientific researcher, he has a strong research interest in the area of stochastic operations research, most notably queuing and simulation. Beyond his scientific activities, he has also taken up the mission of popularizing the potential of OR for business environments and general public. Accordingly, he has written articles for Dutch magazines and national newspapers.

He has supervised and been involved in a variety of practical projects, among which for the Dutch railways, the Dutch airport, the Dutch Triple A, the Dutch ministry of health, hospitals and industry.

1 Introduction

Simulation or more precisely as meant in the setting of this paper: discrete event simulation is known as a most powerful tool for the evaluation of logistical systems such as arising in manufacturing, communications or the service industry (banks, call centers, hospitals). A general characterization is that these systems:

- are complex
- involve stochastics and
- require some form of optimization (such as by infrastructure, lay-out or work procedures).

Analytic and optimization methods, as standardly covered by the field of OR (Operations Research), in contrast, only apply if:

- the systems are sufficiently simple and
- special assumptions are made on the stochastics involved.

On the other hand, simulation by itself does not provide: insights and techniques for optimization.

If a limited finite number of scenarios is already available or if an optimization problem can be parameterized, different search approaches can be suggested to expedite and automate an optimization (also referred to as *simulation based optimization*). An elegant exposé of such methods can be found in [2].

But otherwise, simulation does not imply an optimization. This is where the discipline of Operations Research (OR) might contribute in either of three directions:

1. To suggest candidate scenarios as based upon OR-results and insights.
2. To provide OR-optimization techniques in addition to simulation.
3. To integrate OR-optimization with simulation.

A combination of OR and simulation might then become most beneficial.

- Simulation for evaluation,
- OR for optimization.

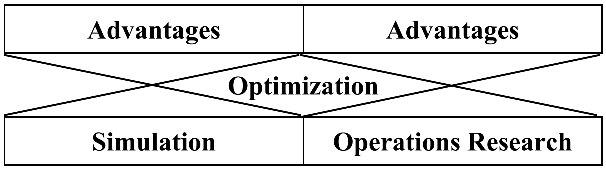
The advantages of this combination for optimization are schematically represented in Tab. 1.

In this paper, each of these three directions of this combination will be illustrated by a specific practical application. Direction

1. by pooling in call centers (see [4], [5])
2. by check-in capacity minimization(see [3])
3. by blood platelet production and optimization (see [1]).

Tab. 1: Combined Advantages

Simulation	**OR**
Advantages	**Disadvantages**
Real-life complexities Real-life stochastics	Simple models Strict assumptions
Disadvantages	**Advantages**
Evaluation By scenarios By numbers only	Optimization By techniques Also by insights



The applications all rely upon recent research. The technical details for each of them can be found in the separate technical papers.

The OR-techniques in these applications involve

- Queueing
- Linear programming
- Stochastic Dynamic Programming

The results, as based upon these practical applications, seem to indicate that this combined Simulation-OR approach can be most fruitful. Further application and research of this approach is therefore suggested.

2 Call centers: To pool or not?

Should we pool servers or not? This seems a simple question of practical interest, such as for counters in postal offices, check-in desks at airports, physicians within hospitals, up to agent groups within or between call centers. The general perception seems to exist that pooling capacities is always advantageous.

An instructive example (Queueing)

This perception seems supported by the standard delay formula for a single (exponential) server with arrival rate λ and service rate μ: $D = 1 / (\mu - \lambda)$. Pooling two servers thus seems to reduce the mean delay by roughly a factor 2 according to $D = 1 / (2\mu - 2\lambda)$.

However, when different services are involved in contrast, a second basic result from queueing theory is to be realized: Pollaczek-Khintchine formula. This formula, which is exact for the single server case, expresses the effect of service variability, by:

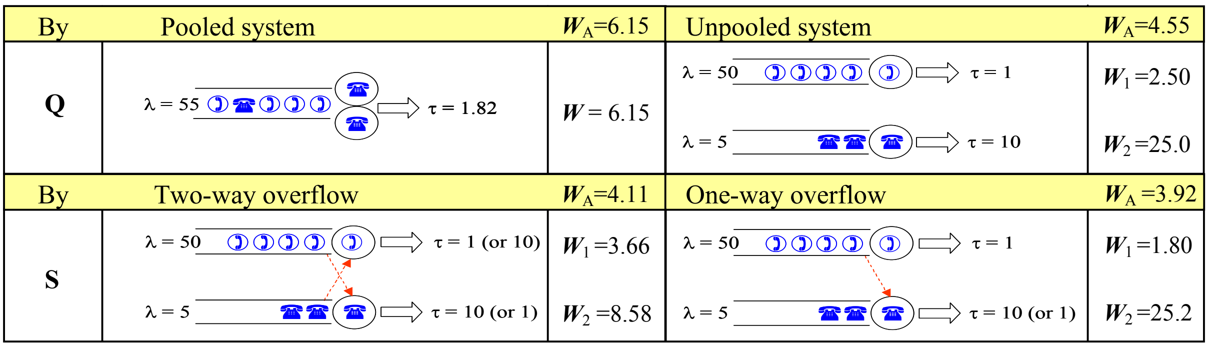


Fig. 1 Pooling Scenarios by Queueing (Q) and Simulation (S)

$W_G = ½ (1+c^2)\, W_E$ with $c^2 = \sigma^2 / \tau^2$ and

W_G the mean waiting time under a general (and E for exponential) service distribution with mean τ and standard deviation σ.

By mixing different services (call types) extra service variability is brought in which may lead to an increase of the mean waiting time.

This is illustrated in Fig. 1 for the situation of two job (call) types 1 and 2 with mean service (call) durations $\tau_1 = 1$ and $\tau_2 = 10$ minutes but arrival rates $\lambda_1 = 10\ \lambda_2$. The results show that the unpooled case is still superior, at least for the average waiting time W_A. Based on these queueing insights, a two-way or one-way overflow scenario can now be suggested, which leads to further improvement as also illustrated in Fig. 1.

A Combined approach

To achieve these improvements simulation is necessarily required. A combination of queueing for its insights to suggest scenarios and of simulation for evaluating these scenarios thus turns out to be fruitful.

Rank	No.	Scenario	W_A
7	1	Pooled	0.71
6	5	Thr(Opt)	0.68
5	2	Unpooled	0.63
4	3	Two-way	0.58
3	4	One-way	0.52
2	6	Prio(1,NP)	0.38
1	7	Prio(1,P)	0.20

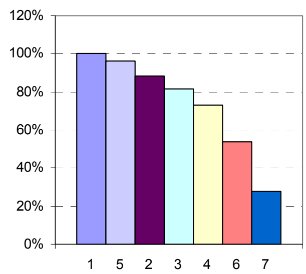


Fig. 2 Average waiting times for different scenarios

Call centers (large number of servers)

Similar results can also obtained for larger number of servers, say with 10, 50 or 100 servers, such as arising in realistic call centers. The one-way overflow scenario turns out to be superior to both the pooled and the unpooled scenario for realistically large numbers of call centers agents. (Here the mix ratio of short and long services is similar as in the example above. For further details, see [4].) As there are no analytic solutions for queueing systems with overflow, these results necessarily had to be obtained by simulation.

In fact, as shown in [5] by OR (queueing) insights substantial improvements (see Fig. 2) can still be achieved by practical rules that appear to be nearly optimal.

3 Check-in planning

Problem formulation and combined approach

Check-in desks and desk-labor hours can be a scarce resource at airports. To minimize the required number of desks two essentially different optimization problems are involved:

P1: A minimization of the required number of desks for a given flight.
P2: A minimization and scheduling for all flights during a day.

A two-step procedure is therefore proposed:

Step 1: For P1 as based upon simulation
Step 2: For P2 as based upon Linear Programming

Step 1: Simulation

As the check-in process is highly transient (fixed opening interval, non-homogeneous arrivals during opening hours and initial bias at opening time) transient (or terminating) simulation will necessarily be required.

In step 1 therefore the required number of desks will have to be determined by terminating simulation for each hour of the day and separate (group of) flights.

Step 2: OR (Linear programming)

Next in step 2 the desks are to be scheduled for the different flights so as to minimize the total number of desks and desk-(labor)-hours. Here additional practical conditions may have to be taken into account such as most naturally that desks for one and the same flight should be adjacent.

Example

As a simple (fictitious) example consider the desk requirements for 5 flights during 9 hours (periods), as determined by step 1. The total number of desks required then never exceeds 4.

flight	1	2	3	4	5
starting period	1	3	4	5	7
ending periode	3	5	6	7	9
# desks required	3	1	2	1	3

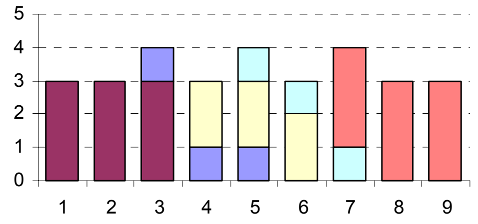


Fig. 3 Desks Requirements of 5 Flights

However, a straightforward Earliest Release Date (ERD) desk allocation as shown in Fig. 3 would lead to an unfeasible solution, as the desks for flight 5 are not adjacent. (This could be resolved by using two more desks 5, 6 and assigning desks 4, 5 and 6 to flight 5). However, in this example a feasible solution with 4 desks is easily found.

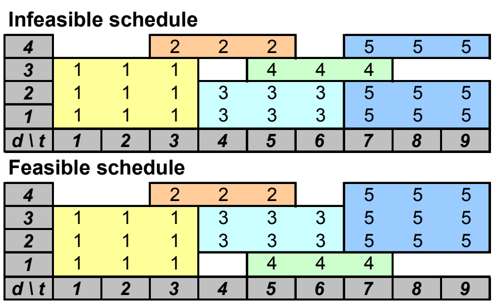


Fig. 4 An (in)feasible schedule

As shown in [3] also for more realistic orders with hundreds of flights an optimal solution can be found by solving an LP-formulation as given below.

$$\begin{aligned}
&\min \quad D \\
&s.t. \quad n_f \le d_f \le D \qquad \forall f \\
&\qquad \left.\begin{array}{l} d_f + n_g \le d_g \text{ or} \\ d_g + n_f \le d_f \end{array}\right\} \quad \forall f,g \text{ with } I_f \cap I_g \ne \varnothing
\end{aligned}$$

where

D: Total number of desks required (indexed 1 to D);
I_f: Check-in time interval of flight f (with f=1,…,F);
d_f: Largest desk number assigned to flight f;
n_f: Number of desks required for flight f.

Also shown in [3] a similar LP-formulation can also be given for the optimization problem of variable allocation in which the number of desks, as determined by step 1, may vary by the hour which may lead to further savings. This is illustrated in Fig. 2 for an example data set of 10 flights which leads to a further reduction in desks (from 17 to 15) and desk hours (from 117 to 92). The combination of (terminating) simulation and LP-optimization so turned out to be most beneficial.

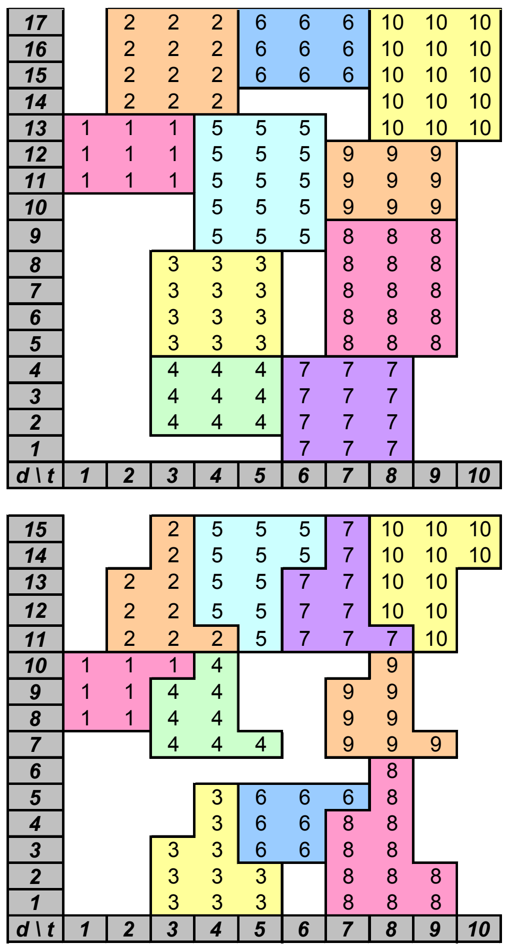


Fig. 5 Feasible and infeasible schedule

4 Blood management

4.1 Motivation

Blood management is a problem of general human interest with a number of concerns and complications. This supply has to rely upon voluntary donors. As a complicating factor, blood platelets (thrombocytes) have a limited life-time of at most 6 days.

Clearly, as lives may be at risk, shortages are to be minimized. On the other hand, as the supply is voluntary, blood is to be considered as highly precious. Also spill by outdating, of blood platelets is thus to be minimized.

4.2 OR - Simulation approach

In [1] a combined 'new' approach for the blood platelet inventory problem has therefore been followed, which integrates an OR-technique with simulation by the following steps:

- Step 1: First, a stochastic dynamic programming (SDP) formulation is provided.
- Step 2: The dimension of the (SDP) formulation is then reduced (downsized) by aggregating the state space and demands so that the downsized (SDP) problem can be solved numerically (using successive approximation).
- Step 3: Then, as essential tying step, the optimal policy for the downsized SDP is (re)evaluated and run by simulation in order to investigate the structure of the optimal strategy.
- Step 4: By the structure at the 'simulation table' a simple practical near to optimal order-up-to strategy is then derived.
- Step 5: The quality (near-to-optimality) of this practical simple order-up-to strategy is then also evaluated by simulation.

As the technical (mathematical) details of steps 1 and 2 are somewhat 'standard' but also 'complicated' and worked out in detail in [1], let us just restrict to an illustration of the essential OR and Simulation steps 1 and 3.

4.3 OR step 1

As for step 1 the state of the system is described by $(d, \boldsymbol{x})$ with

d: the day of the week (d = 1,2,….,7) and

$\boldsymbol{x} = (x_1, x_2,…, x_m)$ the inventory state with x_r = the number of pools with a residual life time of r days (maximal m = 6 days) (A pool is one patient-transfusion unit containing the platelets of 5 different donations).

And let

$\boldsymbol{V}_n(d,\boldsymbol{x})$: represent the minimal expected costs over n days when starting in state $(d,\boldsymbol{x})$.

The optimal inventory strategy and production actions are then determined by iteratively computing (solving) the SDP-equations for n = 1,2,…..

$$\boldsymbol{V}_n(d,\boldsymbol{x}) = \min_k \left[c(\boldsymbol{x},k) + \sum p_d(b)\, \boldsymbol{V}_{n-1}(d+1, t(\boldsymbol{x},k,b))\right]$$

with

k the production action,
$c(\boldsymbol{x},k)$ the one day costs in state $\boldsymbol{x}$ under production k,
$p_d(b)$ the probability for a (composite) demand b,
$t(\boldsymbol{x},k,b)$ the new inventory state depending on k, b, $\boldsymbol{x}$, and some issuing policy, and
$\boldsymbol{V}_0(d, \boldsymbol{x}) \equiv 0$.

4.4 Simulation step 3

However, in practice one needs a simple rule and this optimal strategy has no simple structure. In order to derive a simple order-up-to strategy which only depends on the total predicted inventory, the actual platelet production-inventory process is therefore simulated for 100,000 replications so as to register how often which total predicted final inventory level (I) and corresponding action occurs under the optimal strategy (as determined by SDP). As an illustration, for a particular day of the week (in this case Tuesday) and the dataset of the regional blood bank, this led to the '*simulation table*' in Tab. 2.

For example, it shows by row 15 and column 7 that during the 100,000 replications 2593 times a state was visited with a total expected final inventory (I) of 7 followed by a production decision of 8 (order-up-to 15). This order up-to-level of 15 occurs in 74.5% of the states visited and can be seen as a target-inventory level for Wednesday mornings.

Tab. 2: Simulation Frequency table of (State, Action)-pairs on Tuesdays from Simulation of Optimal SDP Solution for 100,000 weeks

I	2	3	4	5	6	7	8	9	10	11	12	13	14	cum.
Order-up-to														
23													4	4
22												28		28
21											96			96
20										267				267
19								2	748	3				753
18							18	1928	31	1				1978
17						6331	4490	353	26	1				11201
16					8260	2078	783	7						11128
15	3131	14123	20926	23646	10087	2593	39							74545
:														
0														
cum.	3131	14123	20926	23646	18347	11002	5330	2290	805	272	96	28	4	100000

4.5 Results

Applying this approach to data from a Dutch regional blood bank, we could draw the following conclusions:

1. The simple order-up-to rule reduces the spill from roughly 15 to 20%, as a figure that also seems rather standard worldwide, to less than 1% (while also shortages were reduced and nearly vanished).
2. The combined SDP-Simulation approach led to accuracy within 1% of the exact optimal value for the downsized problem.

For the detailed results, we refer to [1].

5 Evaluation

Simulation is standardly used and known for evaluation purposes of process performance. Its application for optimization purposes, however, seems to be limited mainly to a comparison of scenarios or parameterized search methods as in [2].

In this paper in contrast, it is illustrated that simulation can also be used in a more sophisticated way in combination with OR-techniques.

To this end, three illustrations are provided in each of which simulation is used in combination with a different OR-technique. The OR techniques involve:

- Queueing
- Linear Programming
- Stochastic Dynamic Programming (SDP)

The results, as based upon different practical applications, seem to indicate that this combined Simulation-OR approach can be most fruitful. Further application and research of this approach is therefore suggested.

6 References

[1] Haijema, R., J. van der Wal and N. M. van Dijk (2006) Blood Platelet Production: Optimization by Dynamic Programming and Simulation. *Computers and Operations Research* 34, 760-779.

[2] Krug, W. 2002. Modelling, Simulation and Optimisation for Manufacturing, Organisational and Logistical Processes. Erlangen, Deutschland, Gruner Druck GmbH.

[3] Van Dijk, N. M., and E. van der Sluis (2006). Check-in computation and optimization by simulation and IP in combination. *European Journal of Operations Research* 171, 1152-1168.

[4] Van Dijk, N. M., and E. van der Sluis (2007). To pool or not to pool in Call Centers. *Production and Operations Management*, to appear.

[5] Van Dijk, N. M., and E. van der Sluis (2007). Pooling is not always the answer. *European Journal of Operational Research* (in revision).

