16th European Symposium on Computer Aided Process Engineering
and 9th International Symposium on Process Systems Engineering
W. Marquardt, C. Pantelides (Editors)



# Simulation based optimization for risk management in multi-stage capacity expansion

Xiaotao Wan,[a] J. F. Pekny,[b] G. V. Reklaitis[b]

[a]*GE(China) R&D Center Co. Ltd, 1800 Cailun Road, Shanghai, 201203, P.R. China*
[b]*School of Chemical Engineering, Purdue University, West Lafayette, IN, 47906, USA*

## Abstract

Risk management for multi-stage capacity expansion optimizes average return and risk simultaneously. None of the existing algorithms for stochastic dynamic programming can accommodate general risk measures. Algorithms based on simulation based optimization are proposed in this research to address arbitrary risk measures for multi-stage risk management in capacity expansion. These algorithms utilize multi-stage a back-propagation scheme and function approximation techniques. Their effectiveness is demonstrated by applying them to a pharmaceutical product pipeline case study.

**Keywords**: simulation based optimization, dynamic risk management, stochastic dynamic programming, multi-stage capacity expansion.

## 1. Introduction

Strategic capacity decisions are positioned at the top of the hierarchy of supply chain management decisions. Under the risk management framework, the expected return of a capacity decision is simultaneously optimized with the risk of the decision weighted by a risk aversion parameter, where the risk is quantified in terms of variance, semi-norm, value-at-risk etc. It has been proven that stochastic math programming is limited to non-decreasing risk measures, and not applicable to general risk measures (Takriti and Ahmed, 2004). Cheng et al. (2003, 2004a, 2004b) solve the multi-stage risk management problem in capacity expansion through exploring the special property of a separable risk measure; However, the back-propagation scheme for dynamic programming used therein cannot accommodate non-separable risk measures. In this research, new techniques based on simulation based optimization are proposed to address multi-stage capacity expansion problems for arbitrary risk measures and are applied to a pharmaceutical product pipeline case study.

## 2. Algorithms for risk management in dynamic optimization main text

The proposed algorithms extend the Bellman equation of stochastic dynamic programming. Two underlying fundamental components are multi-stage back-propagation and function approximation of simulation results. For a dynamic problem with T stages, let $s$ be the state space, $x$ be the decision, $\omega$ be the random event, a pseudo-utility function is defined as follows:

$$\mathrm{U}(s_t) = \max_x \left[\mathrm{E}(f(s_\tau, x, \omega_\tau) + \mathrm{V}(s_{\tau+1}(s_\tau))) - \lambda \mathrm{R}(s_\tau, x)\right] \tag{1}$$

where $\mathrm{E}(f(s_\tau, x, \omega_\tau) + \mathrm{V}(s_{\tau+1}(s_\tau)))$ is exactly as in the Bellman equation with $f(\cdot)$ being the immediate return and $\mathrm{V}(\cdot)$ being the value function, $\mathrm{R}(s_\tau, x)$ denotes the risk of making decision $x$ at state $s_\tau$ and $\lambda$ is the risk aversion parameter. The pseudo-utility

function is a natural extension of the value function to incorporate risk associated with each state. It represents the desirability of states in the risk management context.

In the Bellman equation, the value function at stage *t* can be obtained by observing the value function at stage *t+1*, through the typical one stage back-propagation scheme. However, the pseudo-utility function at stage *t* generally cannot be calculated this way unless the risk measure in Eq. (1) is separable (Li, 1990). For non-separable risk measures, calculation of the pseudo-utility function at stage *t* requires all the information at the final stage T. The corresponding computational scheme is called multi-stage back-propagation in our study in view of the one stage scheme used to solve the Bellman equation.

For most capacity expansion problems, it is impossible to solve Eq. (1) analytically: simulation based optimization (Fu, 2002) is the only feasible method. At each state in the simulation, a stochastic optimization problem (Eq. (1) ) must be solved to obtain the optimal actions. As the number of states is usually extremely large, function approximation is necessary to mitigate the prohibitive computational burden. The strategy is to generalize the observed simulation results of a sample to the whole space by building an approximation function using the observed sample. Such a strategy has been used to approximate the value function in the Bellman equation to overcome the curse-of-dimensionality (Bertsekas and Tsitsiklis, 1996).

*2.1. Revised back-propagation algorithm*

The pseudo-utility function in Eq. (1) is not amendable to function approximation as no capacity decisions can be derived from knowledge of the function. For purpose of reducing the computational difficulty, we define a state-action pseudo-utility function as

$$\mathrm{U}(s,x)=\mathrm{Q}(s,x)-\lambda\mathrm{R}(s,x) \tag{2}$$

where $\mathrm{Q}(\cdot)$ is the expected return for making decision $x$ at state $s$. The optimal decision for a state $s$ can be found through

$$x^{*}=\arg\max_{x}\mathrm{U}(s,x) \tag{3}$$

Define a random vector $\Psi_{\tau}=(\omega_{\tau},\cdots,\omega_{\mathrm{N}})$, i.e. $\Psi_{\tau}$ represents the random events that occurred between $\tau$ and T. Then the revised back-propagation algorithm for dynamic risk optimization is as follows: at the decision time $\tau$

1. Sample $m$ state-action pairs and obtain $(s_{z},x_{z}),z=1,\cdots,m$.
2. For each pair $(s_{z},x_{z})$, generate realization $\Psi_{i},i=1,\cdots,n$; simulate the $n$ realizations from $\tau$ to T. For each realization, at the decision point $\tau'>\tau$, given the state is $s_{\tau'}$, take action $x_{\tau'}$ according to

$$x_{\tau'}=\arg\max_{x}\widetilde{\mathrm{U}}_{\tau'}(s_{\tau'},x) \tag{4}$$

Where $\widetilde{\mathrm{U}}_{\tau'}(s_{\tau'},x)$ is the approximated state-action pseudo-utility function for time $\tau'$. After compiling the results of the $n$ simulations, compute the expected return $\mathrm{Q}_{\tau}(s_{z},x_{z})$, the risk $\mathrm{R}_{\tau}(s_{z},x_{z})$, and the pseudo-utility $\mathrm{U}_{\tau}(s_{z},x_{z})$ with

$$\mathrm{U}(s_{z},x_{z})=\mathrm{Q}(s_{z},x_{z})-\lambda\mathrm{R}(s_{z},x_{z}) \tag{5}$$

3. Fit an approximate state-action pseudo-utility function $\widetilde{\mathrm{U}}_\tau(\cdot,\cdot)$ based on the *m* points $\mathrm{U}_\tau(s_{\tau z}, x_{\tau z}), z = 1, \cdots, m$.

The multi-stage back-propagation principle is reflected in step 2 where simulation is conducted from τ to T instead of from τ to τ+1 as in the one stage back-propagation. This algorithm builds T-1 approximation state-action pseudo-utility functions, one for each stage except for stage T. Those approximate functions greatly reduce the computational overhead since a deterministic optimization problem represented by Eq. (4) is solved instead of the much more complex stochastic optimization problem represented by Eq. (1) at each state in the simulation.

*2.2. Optimal policy approximation algorithm*

The number of deterministic optimizations in the form of Eq. (4) in the revised back-propagation algorithm is proportional to the number states visited in the simulation, thus the curse-of-dimensionality persists. The following optimal policy approximation algorithm tackles the dimensionality issue with another level of function approximation.

In stochastic dynamic programming, a policy is a function returning optimal actions for any state. Let $\widetilde{\Pi}_\tau(s)$ denote the approximation optimal policy at the decision time τ, then for the decision time τ

1. Sample m state-action pairs and obtain $(s_{\tau z}, x_{\tau z}), z = 1, \cdots, m$.
2. For each pair $(s_{\tau z}, x_{\tau z})$, generate realization $\Psi_{\tau i}, i = 1, \cdots, n$; simulate the n realizations from τ to T. For each realization, at the decision point τ'>τ, given the state is $s_{\tau'}$, take action $x_{\tau'}$ according to

$$x_{\tau'} = \widetilde{\Pi}_{\tau'}(s_{\tau'}) \tag{6}$$

After compiling the results of the n simulations, compute the expected return $\mathrm{Q}_\tau(s_{\tau z}, x_{\tau z})$, risk $\mathrm{R}_\tau(s_{\tau z}, x_{\tau z})$, and pseudo-utility $\mathrm{U}_\tau(s_{\tau z}, x_{\tau z})$ using

$$\mathrm{U}(s_{\tau z}, x_{\tau z}) = \mathrm{Q}(s_{\tau z}, x_{\tau z}) - \lambda \mathrm{R}(s_{\tau z}, x_{\tau z})$$

3. Fit an approximate state-action pseudo-utility function $\widetilde{\mathrm{U}}_\tau(\cdot,\cdot)$ based on the m points $\mathrm{U}_\tau(s_{\tau z}, x_{\tau z}), z = 1, \cdots, m$.
4. For each state $s_{\tau z}, z = 1, \cdots, m$, find its optimal action $x_{\tau z}(s_{\tau z})$ via

$$x_{\tau z}(s_{\tau z}) = \arg\max_x \widetilde{\mathrm{U}}_\tau(s_{\tau z}, x) \tag{7}$$

Fit an approximation optimal policy function $\widetilde{\Pi}_\tau(s)$ with the m points $(s_{\tau z}, x_{\tau z}(s_{\tau z}))$, *z*=1,⋯,*m*.

In this algorithm, T-2 approximate functions for optimal actions are built from stage 2 to state T-1. The number of deterministic optimization performed in the form of Eq. (7) is O(T×m), which is independent of the number of states visited in the simulation. As a result, the curse-of-dimensionality is avoided. In our research, least squares support vector machine (LSSVM) (Wan et al., 2005) is adopted to build all the approximation functions.

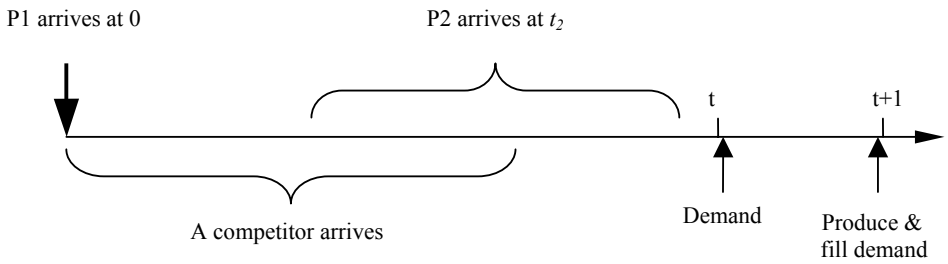


Figure 1: Scheme of the case problem

## 3. Risk management in pharmaceutical capacity expansion

When a pharmaceutical company expands its manufacturing capacity upon new drugs exiting its development pipeline, it may increase the capacity just enough to meet the forecasted demand or it may purchase more capacity for future drugs to reduce the setup cost. However, purchasing capacity for future drugs inevitably incurs risk: the capacity may not match the demands of the future drugs, extra-capacity will reduce the return of investment, while a capacity shortage will necessitate another purchase which incurs an additional undesirable setup cost. The uncertain exit time of future drugs may also make it cost effective to perform additional capacity expansion. Other important factors include competitors: there exists the possibility that competitors will enter the market in the future to take away part of the demand. The right capacity level can only be identified through solving a multi-stage risk management problem.

### *3.1. Case Study: Capacity expansion in a pharmaceutical company*

A pharmaceutical company A has a new drug (P1) exiting its development pipeline at the beginning of the horizon, and the initial available capacity is 0. The demand for the drug is stationary, following a normal distribution N (20, 9) in each period. The total horizon considered is 40 periods. Within the horizon and with a probability 0.5, a second new drug (P2) will exit the pipeline. The exit time follows a triangular distribution Tri (10, 20, and 30). The demand for the second drug is also stationary; with normal distribution in each period with mean N (20, 9) (i.e. the mean demand is uncertain) and the coefficient of variation the same as that of the first drug. Assume the second drug is similar to the first drug: they have the same production cost, market price, etc. A single competitor B exists whose product will share the demand of the first drug if it enters the market but does not affect the demand of the second drug. The arrival of B's product follows an exponential distribution with expected arrival time 45. If B enters the market, its product will take away normal distributed market share N (0.4, 0.01) from A's first product.

### *3.2. Implementation of the optimal policy approximation algorithm*

As shown in Fig. 1, this case problem is a dynamic optimization problem with capacity decision at 0 and $t_2$, and contingent production decisions at each period. In accordance with the proposed optimal policy approximation algorithm, the problem is approached as follows: sample the state-action space at $t_2$, build the state-action pseudo-utility function and consequently the state-optimal action function (i.e. the policy) after simulating the sampled points; sample the state-action space at 0, simulate and build the corresponding state-action pseudo-utility function; obtain the optimal capacity decision at 0 by optimizing the corresponding state-action pseudo-utility function. The default number of sampled points is 40 for the first stage and 650 for the second stage; the

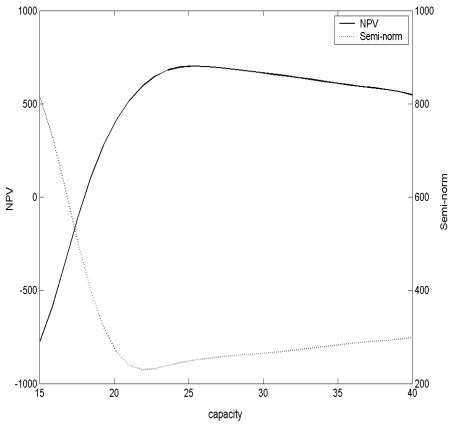


Figure 2: NPV and semi-norm of the first stage capacity decision

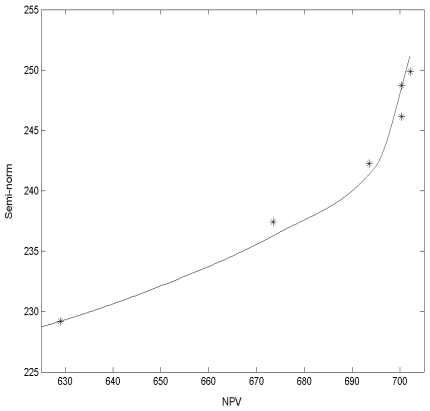


Figure 3: NPV and semi-norm efficient frontier

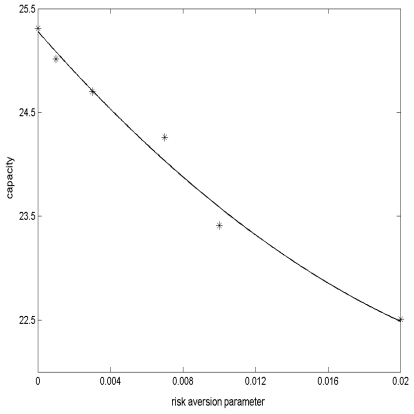


Figure 4: Optimal first-stage capacity decisions under different risk aversion parameters

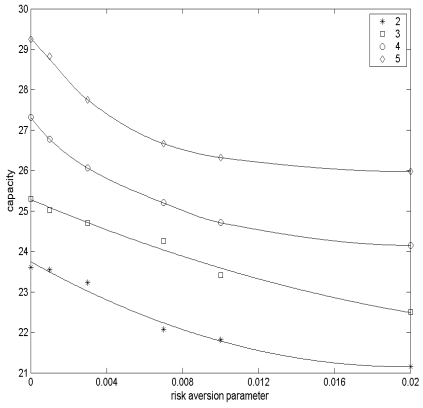


Figure 5: The effect of the demand variance on first-stage capacity decisions

default number of sample path simulated for each sampled point is 4000. Those default numbers are chosen such that they provide satisfactory results for this case, and there is no significant improvement with larger values.

The implementation of the revised back-propagation algorithm is similar to the above procedure except that the state-optimal action surrogate model is not constructed and a deterministic optimization problem is solved at $t_2$ while simulating a sample path from 0 to T.

*3.3. Results and discussion*

For the non-separable risk measure semi-norm, Fig. 2 shows that the optimal decisions for NPV and semi-norm are 26 and 22 respectively when $\lambda$ is equal to 0.001, indicating

that the optimal decision for pseudo-utility must lie between 26 and 22 to balance their trade-off. For other values of $\lambda$, there will be similar relations between the first stage capacity and the NPVs as well as the semi-norms, from which the corresponding optimal decisions together with the NPVs and semi-norms under these decisions can be calculated. The results are presented as the efficient frontier in Fig. 3. This figure simply states that capacity expansion under dynamic conditions demonstrates the NPV and risk trade-off, well known in stock portfolio management: higher NPVs are necessarily associated with higher risks.

Fig. 4 shows that the optimal first-stage capacity level decreases as the risk aversion parameter increases to avoid the risk of lower demand either due to possible arrival of the competitor or non-materization of expected future drugs. Fig. 5 studies the effect of demand variance under different risk aversion parameters on the first-stage capacity level. Clearly, larger demand variances lead to higher capacity levels to avoid the cost of missing demand.

## 4. Conclusions

Two algorithms are proposed for risk management in dynamic optimization based on multi-stage back-propagation scheme and function approximation. The algorithms are the first of their kind to be valid for arbitrary risk measures. Their effectiveness is illustrated by computing the NPV vs. risk efficient frontier for a dynamic capacity expansion case problem.

## References

Bertsekas, D. P., Tsitsiklis, J. N., 1996. Neuro-dynamic programming. Athena Scientific.

Cheng, L., Subrahmanian, E., Westerberg, A. W., 2003. Design and planning under uncertainty: issues on problem formulation and solution. Comp. Chem. Engng. 27, 781–801.

Cheng, L., Subrahmanian, E., Westerberg, A., 2004a. A comparison of optimal control and stochastic programming from a formulation and computation perspective. Comp. Chem. Engng 29(1).

Cheng, L., Subrahmanian, E., Westerberg, A., 2004b. Multi-objective decisions on capacity planning and production-inventory control under uncertainty. Ind. Eng. Chem. Res. 43, 2192–2208.

Li, D., 1990. Multiple objectives and non-separability in stochastic dynamic programming. International Journal of System Science 21 (5), 933–950.

Fu, M. C., 2002. Optimization for simulation: theory vs. practice. INFORMS Journal on Computing 14 (3), 192–215

Takriti, S., Ahmed, S., 2004. On robust optimization of two-stage systems. Mathematical Programming 99, 109–126.

Wan, X., Pekny, J. F., Reklaitis, G.V., 2005. Simulation-based optimization with surrogate models—application to supply chain management. Comp. Chem. Engng 29 (6), 1317-1328