

# Reverse-Simulation 기법에 의한 다수 평가 함수를 가진 시스템의 최적화

박 경중†, 이 영해‡, 박 영석§

† 동일 CIM, ‡ 한양대학교 산업공학과, § 경일정보기술

## Abstract

Simulation is commonly used to find the best values of decision variables for problems which defy analytical solutions. "Simulation Optimization" technique is used to optimize the system design problems which can not be expressed in analytical of mathematical models.

In this research, we will study Reverse-Simulation optimization method which is quite different from current simulation optimization methods in literature. We will focus on the on-line determination of steady-state method which is very important issue in Reverse-Simulation optimization, and the construction of Reverse-Simulation algorithm with expert systems.

Especially, in the case of multiple objectives because of the dependency of simulation model, all objectives do not satisfied simultaneously. In this paper, therefore, we process simulation optimization using objectives with priority to optimize multiple objectives under single run.

## 1. 서론

현대의 많은 복잡한 실제 시스템은 이산형 시스템으로 모델링되며, 이들 시스템들은 일반적으로 이산 사건의 발생과 시간에 따른 상태변화에 의해 유도된다. 이와 같은 이산 사건의 복잡한 관계는 확정적 추론 기법(deterministic approximation techniques)이나 확률적 시뮬레이션(stochastic simulation)에 의해 연구되어야 한다.

그러나, 도착 시간과 서비스 시간의 랜덤 변수들이 임의의 분포, 제한된 버퍼 크기, 우선순위 등을 가정하는 모델들은 분석적인 방법으로 해결할 수 없다. 따라서 이러한 모델들은 시뮬레이션 기법으로 해결해야 한다.

시뮬레이션 모델링은 제조 설비와 같은 복잡한 확률적 시스템을 분석하고, 수행 척도 등을 계산하기 위해 광범위하게 사용되고 있으며, 우리는 제어 가능한 파라미터들에 대한 시스템의 수행 척도를 최적화하는데 관심이 있다. 시뮬레이션 모델의 최적값을 찾기 위한 전통적인 방법들은 계산량이 많은 시뮬레이션 반복을 필요로 하기 때문에, 지금까지의 시뮬레이션 최적화 방법들은

은 시뮬레이션의 반복 수를 줄이면서 최적값을 찾는 곳으로 가고 있다고 해도 과언은 아니다.

시뮬레이션 최적화란 시뮬레이션 방법을 사용하는 확률적 시스템에 대한 최적화 연구이며, 확률적 최적화의 일부로서 설명될 수 있다. 시뮬레이션을 사용한 평가는 지금까지 What if 문제에 대응하는 형태를 나타내었으나, 최근 수년 동안 컴퓨터 시뮬레이션의 성공이 How to 질문에 답을 할 수 있는 정도로 확대되었다. What if 질문은 시스템의 결정 변수들에 대한 주어진 일련의 값의 수행척도에 대한 답을 주며, How to 질문은 시스템의 결정변수에 대한 최적값을 찾아주므로 주어진 응답치나 응답 벡터가 최대화되거나 최소화된다. 지금까지는 시스템의 정량적인 결정변수의 최적화를 다루는 시뮬레이션 최적화 절차가 행해졌으며, 복잡한 시스템 구조의 최적화를 다루는 절차에 대한 연구가 계속되고 있다. 시뮬레이션 최적화에 대한 연구는 Ho 와 Cao[5], Glynn[3], Meketon[9], Jacobson 과 Schruben[6], Safizadeh[13]에 의해서 발전되었다.

그러나, 기존의 시뮬레이션 최적화 방법은 많은 문제점을 가지고 있기 때문에, 시뮬레이션 최적화를 위해 새로운 기법인 단일 실행 Reverse-Simulation 기법에 대해서 논한다. 특히 Reverse-Simulation 기법을 사용하여 목표치가 2 개 이상인 경우에 동시에 모든 목표치들이 만족되지 못할 때, 단일 실행하에서 목표치들을 만족시키기 위한 방안을 제시한다.

## 2. Reverse-Simulation 기법

분석가는 시뮬레이션 실험을 설계하기 위해 시뮬레이션 실험 과정에서 조작되는 시스템 설계 변수들의 초기 환경을 제공해야 한다. 시뮬레이션에 관한 기존 연구에서는 이러한 초기 환경을 결정하기 위해서 직관(intuition)을 이용한 방법, 비용 제약식(cost constraints) 방법, 또는 단순히 추측을 이용한 방법이 제안되었다. 시스템을 모델링하고, 수행도를 평가하고 바람직스러운 수행도가 성취될 때 까지 반복적으로 시스템의 파라미터를 조절하는 기존의 방법 대신에

Reverse-Simulation 기법은 이러한 과정을 역으로 수행하는 방법을 제시하였다[16].

Reverse-Simulation 기법은 1991년에 Wild와 Pignatiello[15]에 의해 처음으로 제안되었으며, Kwanjai와 Wild[7], Wild와 Jackson[14]를 거쳐 1994년에 Wild와 Pignatiello[16]에 의해 개념이 구체화되었다.

Reverse-Simulation 기법의 개념은 “바람직스러운 수행도의 목표값(target values)이나 값의 범위를 결정하고, 이러한 사용자 정의(user-defined) 수행도 목표를 만족시키기 위해 동적으로 시스템 설계를 조정”하는 것이다. Reverse-Simulation은 시스템 수행도 평가와 최적화에 대한 시스템 파라미터를 찾기 위한 시스템 설계 변수들의 가능해를 결정하는데 필요한 유용한 정보를 제공한다. 즉, Reverse-Simulation을 이용하여 시스템 수행도 기준을 만족시키는 시스템 설계 변수들에 대한 환경을 구축할 수 있다.

분석가는 Reverse-Simulation을 위해서 출력 데이터를 미리 예측할 필요가 없지만, 좋은 시스템 수행도가 무엇이고 수행 척도의 바람직스러운 목표 값이나 범위를 미리 규정해야 한다. Reverse-Simulation 기법은 정해진 수행도 목표들에 대해서, 시물레이션 프로그램의 수행 동안 동적으로 시스템 설계 변수 값들을 조작 함으로서 설정된 목표와 제약식을 만족하는 시스템 파라미터를 결정하는 방법이라고 할 수 있는데 수행 단계는 다음과 같다[16].

단계 1: 수행 척도의 목표 값이나 범위에 따라 시스템 목표를 입력한다.

단계 2: 시스템 설계 변수 값들이 목표 수행도를 만족하도록 동적으로 시물레이션을 실행한다.

단계 3: 조합 과정을 수행해 안정된(stable) 시스템 설계를 만들어 내는 시스템 설계 변수들의 가능해 값들을 출력한다.

단계 4: 최적 또는 최상의 시스템 설계를 찾기 위해 계속적인 시물레이션으로 시스템 설계 변수들의 가능해를 설정한다.

지금까지 설명한 바와 같이 Reverse-Simulation 기법은 필요로 하는 결과를 얻기 위한 시스템 파라미터를 설정하자는 이론이다. 따라서 Reverse-Simulation 기법은 단계 1에서 설명한 것 처럼 먼저 시스템 목표 값이나 범위가 주어져야 한다. 이러한 시스템 목표 값은 1개 또는 2개 이상이 될 수 있다. 단계 2에서 설명한 것 처럼 단계 1에서 시물레이션을 수행하면서, 주어진 시스템 목표치를 만족시키는 시스템 설계 변

수들의 값을 얻는다. 이 때, 단계 2의 시물레이션 실행 단계에서 시스템 목표를 만족하는 시스템 설계 변수들의 집합은 1개 이상 될 수 있다. 단계 3에서는 시스템 목표 값을 만족하는 시스템 설계 변수들의 가능해 값들을 얻어낸다. 그리고 단계 4에서는 단계 3에서 얻은 가능해들 중에서 최적의 대안을 결정한다.

결정 변수에 따라서, Reverse-Simulation 기법은 단일 목표치 문제와 복수 목표치 문제로 분류될 수 있다. 목표 값이나 구간을 만족하는 결정 변수들의 성격은 이산형(discrete) 또는 연속형(continuous)으로 나눌 수 있으며, 본 연구에서는 이산형 결정 변수하에서 복수 목표치 문제를 다룬다.

Reverse-Simulation 기법은 한 번 시물레이션을 수행하여 오프-라인으로 결과를 판단하고, 만족스럽지 못하면 시스템 설계 변수들을 수정하여 시물레이션을 반복하는 기존의 시물레이션 수행 방법이 아닌 시물레이션을 한 번만 수행하면서 동적으로 시스템 파라미터를 결정하는 온-라인 방법이다.

Reverse-Simulation 최적화를 수행하는 동안 온-라인으로 시물레이션 결과가 시스템 목표치를 만족하는 가를 확인하는 방법은 쉽지 않다. 본 연구에서는 Reverse-Simulation을 수행하면서 목표로 하는 시스템 파라미터를 결정하기 위해 시물레이션과 전문가 시스템을 연결한다.

시물레이션과 전문가 시스템을 연결하는 방법, 주어진 목표치를 만족할 때의 문제점, 시물레이션을 종료하는 방법은 이와 박[1], 오[10]의 논문을 참고하고, 본 논문에서는 목표치가 2개 이상인 경우에 단일 실행하에서 해결하는 방법에 관해서만 논한다.

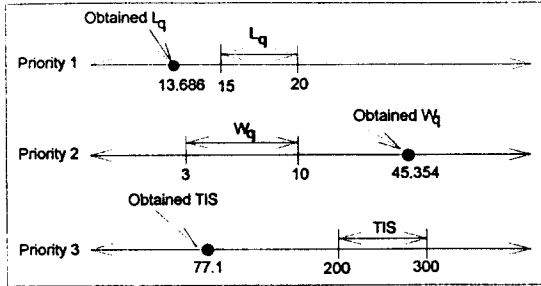
### 3. 다수 평가 함수의 해결 방법

작업물이  $\lambda=18$ 인 지수분포로 도착하여 대기 버퍼에 대기하다가  $\mu=2$ 인 지수분포로 작업을 수행한 후 시스템을 빠져나가는 모델을 고려해 보자. 각각의 대기 버퍼의 용량은 무한대이며, 개체의 작업자 선택 규칙은 선입 선출 방법을 따른다고 가정하자.

이 때의 목표치를 개체의 대기 길이(Lq)를 15~20, 개체의 대기 시간(Wq)을 3~10, 개체의 시스템 체제 시간(TIS)을 200~300으로 유지할 때 필요한 자원의 수를 선택하는 것으로 가정한다.

Reverse-Simulation 최적화 과정을 이와 박[1]에서 제시한 안정 상태와 최적 상태 개

념에 따라 시뮬레이션을 수행하면 다음의 [Figure 1]과 같은 결과를 얻는다. 즉, 시뮬레이션 모델의 의존성 때문에 목표치들이 동시에 모두 만족되지 못한다.



[Figure 1] The results of Reverse-Simulation optimization in 3 objectives

[Figure 1]의 결과를 보면 주어진 목표치 3개 중에서 어느 것도 만족시키지 못하는 결과를 보여준다. 목표치가 2개 이상인 경우에는 목표치들 간에 의존성이 발생하여 목표치들을 모두 만족시키지 못하기 때문에 이런 경우에 해결책이 요구된다. 이런 문제를 해결하기 위한 알고리즘을 다음의 제 3.1 절에서 제시한다.

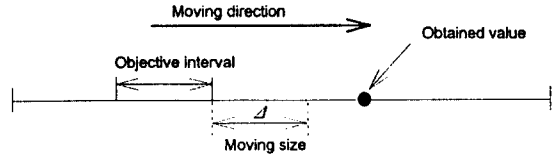
### 3.1 불일치 해결 알고리즘

본 연구에서 제안하는 알고리즘은 다음과 같은 특성을 강화하는데 기본 개념을 둔다. 첫째, 시뮬레이션 모델의 의존성을 최대한 유지한다. 둘째, 단일 실행 최적화가 가능하도록 한다. 셋째, 가장 안 좋은 경우에도 우선순위가 가장 높은 목표치 1개를 만족하는 시스템을 찾는다. 이러한 과정을 설명하면 다음과 같다.

단계 1: Reverse-Simulation 최적화를 수행하여 목표치들의 만족 유무를 판단한다. 모든 목표치들이 만족하면 단계 5로 가고, 그렇지 않으면 단계 2로 간다.

단계 2: 우선순위가 가장 낮은 목표치의 구간을 변경하여 우선순위가 높은 목표치들의 만족 유무를 판단한다. 목표치의 구간을 변경할 때는  $\Delta$  만큼, 시뮬레이션을 수행하여 얻어진 목표치가 존재하는 방향으로 이동하며, [Figure 2]가 이러한 과정을 설명한다.

목표치 구간을 계속해서 바꾸어도 상위의 모든 목표치들이 만족하지 않으면 다음으로 우선순위가 낮은 목표치를 바꾸면서 상위의 모든 목표치들이 만족하도록 한다.



[Figure 2] The moving direction and size of objective value

단계 3: 목표치를 변형시켜서 상위 우선 순위 목표치를 만족시킨 목표치의 값이 변경된 목표치 범위에 들면 나머지 목표치들을 가지고 단계 2로 가고 그렇지 않으면 단계 4로 간다.

단계 4: 얻어진 목표값이 있는 방향으로 이동하면서 만족하는 구간을 설정한다. 이때 상위 목표치들의 만족 조건은 그대로 유지되어야 한다. 충족되지 않은 목표치가 2개 이상 존재하면 단계 2로 가고, 모든 목표치들이 충족되었으면 단계 5로 가고, 그렇지 않으면 단계 4를 반복한다.

단계 5: 시뮬레이션을 종료하고 목표치 구간, 얻어진 값, 그리고 필요한 자원의 수를 출력한다.

### 3.2 수치 실험

본 절에서는 3.1 절에서 설명한 알고리즘에 따라 수치 예제를 들어 설명하며, 이동하는 목표치 구간의  $\Delta$  크기는 주어진 목표치 구간과 같은 크기로 설정한다.

단계 1: 주어진 3개의 목표치들을 가지고 안정 상태와 최적 상태 조건을 찾았으나 만족하지 못하고 종료되었다. 종료시 얻어진 값은 다음과 같다.

$$L_q : 13.686, W_q : 45.354, TIS : 77.1$$

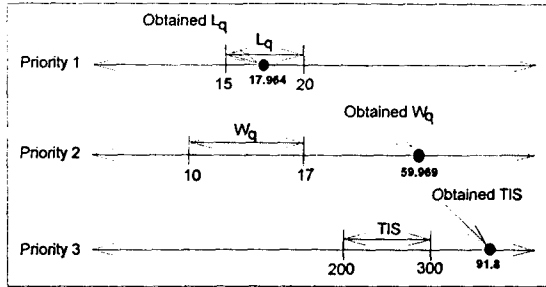
단계 2: 우선순위가 가장 낮은 TIS를 희생시키면서 상위 목표치들의 만족 조건을 찾는다. 얻어진 TIS가 77.1이므로 -TIS 방향으로 구간을 100만큼 이동시켜서 조사한다. 이 때 얻어진 값은 다음과 같다.

$$L_q : 18.027, W_q : 59.609, TIS : 91.4$$

결과를 살펴보면 TIS를 - 방향으로 희생시키면  $L_q$ 와  $W_q$ 가 동시에 증가하여  $L_q$ 만 만족하고,  $W_q$ 는 만족하지 못하므로 TIS의 변동은 의미가 없다. 따라서 TIS를 처음의 상태로 환원하고  $W_q$ 를 희생시키면서  $L_q$ 의 만족 조건을 찾는다.

처음에 얻어진  $W_q$ 가 45.354이므로

$+W_q$  방향으로 7 만큼 구간을 증가시킨다. 이 때 얻어진 값은 다음의 [Figure 3]과 같다.



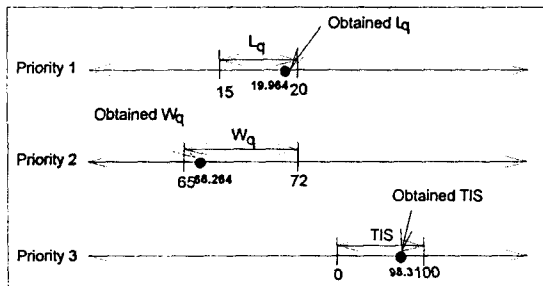
[Figure 3] The results of Reverse-Simulation optimization after step 2

단계 3:  $W_q$ 의 목표치를 3 ~ 10 구간에 10 ~ 17 구간으로 이동하여  $L_q$ 의 값 17.964를 만족하였다. 그러나 이 때 얻어진  $W_q$ 의 값은 59.969 이므로 단계 4로 간다.

단계 4:  $W_q$ 의 구간을 + 방향으로 7 만큼 이동시키면서  $L_q$ 의 만족 구간을 변화시키지 않으면서  $W_q$ 의 변화 구간과 얻어진 값이 일치하도록 하기 위해 시뮬레이션을 수행한다.  $W_q$ 의 구간  $65 \leq W_q \leq 72$ 에서 조건을 충족하며 얻어진 값은 다음과 같다.

$$L_q : 19.516, W_q : 65.173, TIS : 97.2$$

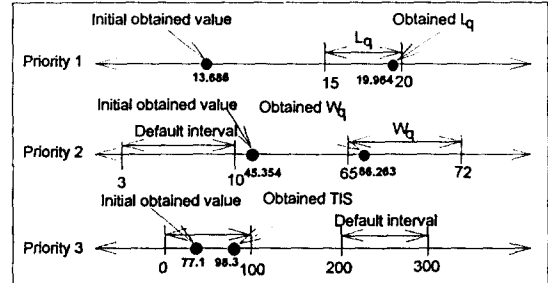
$L_q$ 와  $W_q$  목표치가 만족되고 TIS 목표치 만족 조건만 충족시키면 되기 때문에 -TIS 방향으로 구간을 이동하면서 충족 조건을 탐색한다. 이 때 얻어진 값은 다음의 [Figure 4]와 같다.



[Figure 4] The results of last Reverse-Simulation optimization

단계 5: 모든 조건을 만족하므로 종료한다. 시뮬레이션을 종료한 후 목표치들의 구간 변화를 보면 다음의 [Figure 5]와 같다.

처음에는 목표치들이 만족되지 못하다가 제안된 알고리즘에 따라 Reverse-Simulation을 수행하면 목표 구간이 변경되어 만족되는 영역을 찾을 수 있다.



[Figure 5] The interval change of objective values

변경된 목표치 구간에 대한 이론치를 Hillier[4]와 Little[8]의 식에 따라 구하면 최적 자원의 수가 9가 된다. 따라서 본 연구에서 제시한 알고리즘의 효율성을 검증하기 위해 전체 10번의 시뮬레이션을 수행하여 다음의 [Table 1]을 얻었다.

[Table 1] The results of 3 objectives

#	$L_q$	$W_q$	TIS	Utilization
1	18.613	65.566	93.7	8.0467(9)
2	19.656	65.717	97.9	8.9706(9)
3	19.664	65.692	97.6	8.8987(9)
4	19.661	65.562	97.8	8.9981(9)
5	19.753	65.543	98.2	9.1897(10)
6	19.899	66.621	99.1	9.0104(10)
7	18.566	65.354	93.7	8.1159(9)
8	19.851	65.891	98.3	9.1052(10)
9	20.000	66.544	98.7	8.0228(9)
10	18.628	65.558	93.9	8.1238(9)

[Table 1]의 결과를 보면 목표치를 만족하는 자원의 수가 9와 10의 영역으로 수렴함을 알 수 있으며, 이러한 결과는 시뮬레이션 모델의 확률적 특성 때문에 정확하게 한 개의 값으로는 나타나지 않는다. 얻어진 자원의 수가 이론치 9를 포함하고 있어서, 기존의 Reverse-Simulation 기법의 불확실성을 매우 많이 감소시켜 주고 효율적임을 알 수 있다.

#### 4. 결론 및 추후 연구 과제

기존의 최적화 방법은 사용자의 요구 수준을 고려하지 않은 최적화 방법이었으나, 본 연구에서는 사용자의 요구 조건을 충족할 만큼만 최적화를 수행하는 Reverse-Simulation 방법을 소개하였다.

그러나, 기존의 Reverse-Simulation 방법은 정확한 안정 상태를 찾지 못하므로 이러한 안정 상태를 온-라인으로 찾아주는 새로운 기법의 개발이 필요하여 본 연구에서는 안정 상태와 최적 상태 개념을 확장하여 적용하였다.

특히, 목표치가 2개 이상인 경우에는 시뮬레이션 모델의 의존성 때문에 모든 목표치들이 동시에 만족하지 못하였는데, 이러한 문제를 해결하고자 목표치들에게 우선순위를 부여하여 해결하고자 하였다.

추후 연구 과제는 이산형 결정 변수 뿐만 아니라 연속형 결정 변수를 고려하는 경우에 나타나는 문제점을 해결해야 하고, 양적인 모델이 아닌 질적인 모델을 고려해야 하는 경우에 발생할 수 있는 문제점을 해결하는 것이다.

#### 참고문헌

- [1] 이 영해, 박 경중, 1996, "단일 실행 시뮬레이션 최적화를 위한 Reverse-Simulation 기법", *한국시뮬레이션학회지*, Vol 5, No. 2, pp. 85-94.
- [2] Farhad Azadivar, 1992, "A Tutorial on Simulation Optimization", *Proceedings of the 1992 W. S. C.*, pp. 198-204.
- [3] Glynn, P. W., "Optimization of Stochastic Systems", *Proceedings of 1986 W. S. C.*, pp. 52-59.
- [4] Hillier, F. S. and Lieberman, G. J., 1990, *Introduction to Operations Research*. 5<sup>th</sup> ed., McGraw-Hill.
- [5] Ho, Y. C. and X. R. Cao, 1991, *Perturbation Analysis of Discrete Event Dynamic Systems*. Kluwer Academic Publishers.
- [6] Jacobson, S. H. and L. W. Schruben, 1989, "Techniques for Simulation Response Optimization", *Operations Research Letters*, 8, pp. 1-9.
- [7] Kwanjai, N. N. and Wild, R. H., 1992, "Knowledge-Based Simulation to Assist in System Design Identification", *Proceedings of the 1992 Winter Simulation Conference*.
- [8] Little, J. D. C., 1961, "A Proof for the Queueing Formula:  $L = \lambda W$ ", *Opns. Res.*, 9(3), pp. 383-387.
- [9] Meketon, M. S., 1987, "Optimization in Simulation: A Survey of Recent Results", *Proceedings of 1987 W. S. C.*, pp. 58-67.
- [10] Oh, Hyung Sool, 1996, Output Analysis for Steady-State Simulation Using Chaos Theory. Ph. D. Dissertation, Dept. of I. E., Hanyang Univ.
- [11] O'Keefe, R., 1986, "Simulation and Expert Systems- A Taxonomy and Some Examples", *Simulation*, 46, pp. 10-16.
- [12] Pritsker, A. A. B., 1986, *Introduction to Simulation and SLAM II*, John Wiley & Sons.
- [13] Safizadeh, M. Hossein, 1990, "Optimization in Simulation: Current Issues and the Future Outlook", *Naval Research Logistics*, 37, pp. 807-825.
- [14] Wild, Rosemary H. and He, Jackson, 1994, "An Architecture to Support Reverse Simulation", *Proceedings of the 1994 S. C. S. C.*
- [15] Wild, Rosemary H. and Pignatiello, Joseph J. Jr., 1991, "An Expert System Based Reverse Simulation Technique", *Proceedings of the 1991 S. C. S. C.*
- [16] Wild, Rosemary H. and Pignatiello, Joseph J. Jr., 1994, "Finding Stable System Designs: A Reverse Simulation Technique", *Communications of the ACM*, 35(10), pp. 87-98.