

PSI 기술을 이용한 시뮬레이션과 최적화*

송한식^{***}, 박준철^{***}, 황수진^{***}, 최민철^{***}, 박상일^{***}

A Simulation and Optimization Using the PSI Technology

Song Han Sik, Park Jun Chul, Hwang Su Jin, Choi Min Cheol,

Park Sang Il

Abstract

The advantage of PSI technology in Excel simulation is that it reduces simulation execution times by using polymorphic vector-like computation, which prevents repetitive interpretations of Excel formula and functions and prevents repetitive readings and writings from and into cell. We introduced and showed by examples the use of PSI technology in Excel modeling, simulation and optimization. We also showed that PSI technology enables us to use conventional Solver Add-ins in simulation optimizations. We found a case where PSI technology leads to an erroneous results in IF tests in comparing uncertain computations to 0(Zero) when many uncertain variables are added and subtracted. We showed a PERT simulation case from Albright's text where PSI technology gave wrong results.

I. 서론

엑셀은 현실문제를 모형으로 쉽게 표현할 수 있고(modeling), 또 그것을 읽고 쉽게 이해할 수 있으며, 표현된 모형을 손쉽게 수정하거나 변형할 수 있기 때문에 매우 편리한 도구로서

* PSI : Polymorphic Spreadsheet Interpreter(다중형 해석기) - Frontline System(www.solver.com) 회사에서 개발한 엑셀 스프레드시트 계산방식

** 동아대학교 경영학부 교수

*** 동아대학교 대학원 경영학과

활용되고 있다. 특히, 해찾기(Solver) 기능과 같은 최적화 도구들이 공급되어서 엑셀의 실제 활용이 매우 활발하다.

현실의 문제가 불확실한 변수(uncertain variables)를 포함한 경우에도 엑셀에서 시뮬레이션을 수행하여 그 기대되는 결과를 얻을 수 있다. 최근에는 엑셀에서 시뮬레이션을 손쉽게 수행할 수 있도록 지원해주는 소프트웨어들이 공급되고 있다. 대학의 교과서에서 채택하고 있는 대표적인 예는 @Risk¹⁾와 CrystalBall²⁾을 들 수 있다(Grossman, 2002).

실제 사용과 교육(학습)면에서 본다면 기존의 엑셀 시뮬레이션 소프트웨어는 몇 가지 제한적인 문제점이 있다. 첫째는 시뮬레이션 모형에서의 최적 의사결정을 위해서는 엑셀 시트상의 셀(cell)을 별도의 변수로 지정하고 그 변수들로 새롭게 표현해야 하는 점이다. 이는 사용면에서 보면 매우 불편한 단점이다. 둘째, 실제 적용면에서 보면, 엑셀에서 시뮬레이션 모형에다 최적화 도구를 적용하면 그 소요시간이 지나치게 길다는 단점이다. 엑셀에서 명령 수행이 해석기(interpreter)를 통해서 이루어지기 때문에 반복계산이 요구되는 시뮬레이션 최적화에서는 동일한 수식해석을 반복하는데 많은 시간을 낭비하고 있다. CrystalBall의 경우는 OptQuest 최적화 프로그램이 포함되어 있지만, 기존의 해찾기(Solver)에 비해서 그 사용이 제한적이다(제한조건의 표현이 엑셀 스프레드시트에서 보다 어려운 편이다). 따라서 기존의 최적화 도구(Solver)에서 시뮬레이션 모형을 바로 적용할 수 있는 기능이 필요하였다. 이러한 요구에 부응하여 Frontline System³⁾은 최근 Risk Solver Engine(V7.0)을 공표하였다. 본 논문에서는 Risk Solver Engine을 소개하고 시뮬레이션과 최적화 문제에의 적용에 대한 간단한 소개를 한 다음, Risk Solver Engine의 사용상의 문제점을 발견하여 소개하고자 한다.

II. PSI 기술과 Risk Solver Engine 소개

1. 시뮬레이션 사례

예를 들어서 다음 엑셀시트와 같은 비즈니스 문제를 생각해보자(이 예제는 User Guide p.31에서 가져왔음. 편의상 영문으로 표기된 라벨을 그대로 사용하였음).³⁾ 시장 수요가 낮은 때

1) @Risk는 Palisade사(<http://www.palisade.com>)의 시뮬레이션 프로그램

2) CrystalBall은 Decisioneering사(<http://www.decisioneering.com>)의 시뮬레이션 프로그램

3) 이 예제는 Sam Savage(OR/MS Today, February and April, 2006)의 예시문제를 인용한 Risk Solver Engine User Guide에서 가져왔음.

(Low Demand)는 판매가격 \$10,000에 60,000개가 팔리며, 시장수요가 높을 때(High Demand)는 판매가격 \$8,000에 100,000개가 팔린다고 가정한다. 그 확률은 반반이다. 제조원가(Unit Cost)는 최소 \$6,00, 최대 \$9,00이며 평균적으로(most likely) \$7,50이다. 고정비(Fixed Costs)는 \$30,000으로 가정한다.

이를 평균값에 의존하여 계산하면 순이익(Net Profit)은 다음과 같다.

$$\begin{aligned} \text{Profit} &= \text{Sales} * (\text{Price} - \text{Unit Cost}) - \text{Fixed Costs} \\ &= 80,000\text{개} * (\$9.00 - \$7.50) - \$30,000 \\ &= \$90,000 \end{aligned}$$

<그림 2> 엑셀을 이용한 단순평균에 의한 순이익 계산

	A	B	C	D	E	F	G
1							
2	Financial Results			Market Scenarios			
3					Low Demand	High Demand	Average
4	Sales in units	80,000	Sales units	60,000	100,000	80,000	
5	Price per unit	\$9.00	Price per unit	\$10.00	\$8.00		
6	Unit cost	\$7.50					
7	Fixed Costs	\$30,000			Unit Cost Scenarios		
8					Low	High	Average
9	Net Profit	\$90,000			\$6.00	\$9.00	\$7.50
10							

그러나 이것을 Monte carlo 시뮬레이션으로 계산하면 다른 결과를 얻는다. 다음 <그림2>는 엑셀로 시뮬레이션 한 것이다. 다음은 입력된 수식이다.

$$\begin{aligned} B13 &= \text{Rand} () \\ C13 &= \text{IF}(B13 \leq 0.5, \$E\$4, \$F\$4) \\ D13 &= \text{IF}(B13 \leq 0.5, \$E\$5, \$F\$5) \\ E13 &= \text{Rand} () \\ E11 &= (G9 - E9) / (F9 - E9) \\ F13 &= \$E\$9 + (\$F\$9 - \$E\$9) * \text{IF}(E13 \leq \$E\$11, \\ &\quad \text{SQRT}(E13 * \$E\$11), 1 - \text{SQRT}((1 - E13) * (1 - \$E\$11))) \\ G13 &= C13 * (D13 - F13) - \$B\$7 \end{aligned}$$

<그림 3> 엑셀 시뮬레이션에 의한 순이익 계산

	A	B	C	D	E	F	G
1							
2	Financial Results				Market Scenarios		
3					Low Demand	High Demand	Average
4	Sales in units	80,000		Sales unit	60,000	100,000	80,000
5	Price per unit	\$9.00		Price per	\$10.00	\$8.00	
6	Unit cost	\$7.50					
7	Fixed Costs	\$30,000		Timer		Unit Cost Scenarios	
8				Microsoft Excel	Low	High	Average
9	Net Profit	\$90,000		평균 9.37500037252903E-03 초 걸렸습니다	\$6.00	\$9.00	\$7.50
10				확인	0.50		
12	순번	난수	수요	판매가격	난수	제조단가	순이익
13	1	0.075493143	60000	10	0.570181949	7.60925156	113444.91
14	2	0.564282951	100000	8	0.726392253	7.89038977	-19038.977
15	3	0.768990579	100000	8	0.852455416	8.18516834	-48516.834
16	4	0.50790376	100000	8	0.733545492	7.90499074	-20499.074
1010	998	0.610703983	100000	8	0.592139954	7.64524164	5475.8357
1011	999	0.970179079	100000	8	0.320419222	7.20078578	49921.422
1012	1000	0.098222835	60000	10	0.732068772	7.9019606	\$95,882
1013						평균	\$69,581.00

수식 B13:G13을 B1012:G1012까지 복사한다. 그리고 G1013 = AVERAGE(G13:G1012)를 구한다. 그러면 그 결과 순이익 평균은 \$69,581인데, 이는 앞에서 평균으로 구한 순이익과 사뭇 다르다. 따라서, 이러한 간단한 문제도 시뮬레이션이 필요하며, 시뮬레이션을 통해서 올바른 값을 구할 수 있음을 알 수 있다.)

그러나 이와 같이 시뮬레이션하기 위해서는 엑셀에서 수식(formula)을 모두 6,000번 해석하고(interpret) 계산하고 셀에 입력하여야 한다(위 엑셀시트에서 B~G열을 1,000번 계산하므로). 즉, 엑셀에서의 시뮬레이션 소요시간은 수식을 해석하는 시간과 계산하는 시간, 및 셀에 입출력 하는 시간으로 구분된다. 위의 시뮬레이션 과정에서 소요된 시간 9.37×10^{-3} (초)는 대부분은 셀(cell) 속의 수식을 해석하고, 셀에서 값을 읽어오고 계산결과를 셀에 표시하는데 걸린 시간이다(시뮬레이션 수행을 100회 한 시간의 평균을 구하였음. 한 번의 수행에는 1,000번의 반복계산을 함). 그림에서 "Timer" 명령버튼은 엑셀 Timer 함수를 이용하여 시간경과를 측정하기 위해 추가한 VBA 프로그램에 의한 것이다. 다음에 소개하는 PSI 기

4) 예제의 이러한 오류는 두 개의 종속된 변수(판매량과 판매가격)를 독립된 변수로 취급한데서 비롯된 것이다.

술에 의한 시뮬레이션은 시간이 많이 단축된 것을 볼 수 있다.

2. PSI 기술 (多衆形 해석기)

PSI 기술은 스프레드시트의 수식을 해석해서 계산하는 일종의 해석기(interpreter)이다. 엑셀에서는 계산을 할 때마다 그 수식을 해석을 해야 하지만 PSI 기술은 한번만 수식을 해석하고 그것을 반복해서 사용하는 기술이다. 위의 시뮬레이션에서 본다면 PSI 기술은 B~G열의 계산을 원소가 1,000개인 6개의 벡터(vector) 취급하고 계산한다. 즉, 벡터 원소의 인덱스 i 에 대한 계산을 1,000번 반복하되 그것을 스프레드시트 셀에서 하지 않고 컴퓨터 메모리 상에서 하는 것이다($i=1, 2, 3, \dots, 1,000$).

<그림 4> PSI 기술을 이용한 시뮬레이션에 의한 순이익 계산

	A	B	C	D	E	F	G
1							
2	Financial Results			Market Scenarios			
3				Low Demand	High Demand	Average	
4	Sales in units	60,000		Sales units	60,000	100,000	80,000
5	Price per unit	\$10.00		Price per unit	\$10.00	\$8.00	
6	Unit cost	\$8.65		0.352791108			
7	Fixed Costs	\$30,000				Unit Cost Scenarios	
8					Low	High	Average
9	Net Profit	\$50,713		Timer	\$6.00	\$9.00	\$7.50
10	True Average	\$69,945.96					
11							
12							
13							
14							
15							

PSI 기술을 이용한 시뮬레이션은 다음과 같이 엑셀함수와 PSI 시뮬레이션 함수를 섞어서 사용한다.

D6 = PsiUniform(0,1) : (0,1)의 균등난수를 발생시킨다.

(실제는 시뮬레이션 수행 횟수만큼 벡터로 발생시킨다)

B4 = IF(D6<=0.5, \$E\$4, \$F\$4)

B5 = IF(D6<=0.5, \$E\$5, \$F\$5)

B6 = PsiTriangular(\$E\$9, \$G\$9, \$F\$9) : 삼각분포의 난수를 발생시킨다.

B9 = B4 * (B5 - B6) - B7

B10 = PsiMean(B10) : B9의 평균값을 계산한다.

여기서 PsiUniform(), PsiTriangular()는 불확실 확률 변수를 생성하는 PSI 분포함수이고, PsiMean()은 통계값(statistic)을 구하는 PSI 통계함수이다.

PSI 기술은 여기서 모두 5,000개의 메모리를 사용한다(D6, B4, B5, B6, B9에 각각 1,000개씩). Timer 함수를 이용해서 측정한 결과 PSI에 의한 시뮬레이션 소요시간은 7.81×10^{-4} 초이었다. 이는 엑셀에서 직접 시뮬레이션을 하는 것보다 거의 10분의 1수준으로 감소한 것이다. PSI 기술을 이용할 경우 소요시간의 대부분을 실제 계산에만 투여하기 때문이다. PSI는 시간을 단축하지만 컴퓨터 메모리를 많이 차지한다. 그러나 컴퓨터의 DRAM 칩이 값싸게 공급되면서 PSI 기술을 효과적으로 사용할 수 있다.

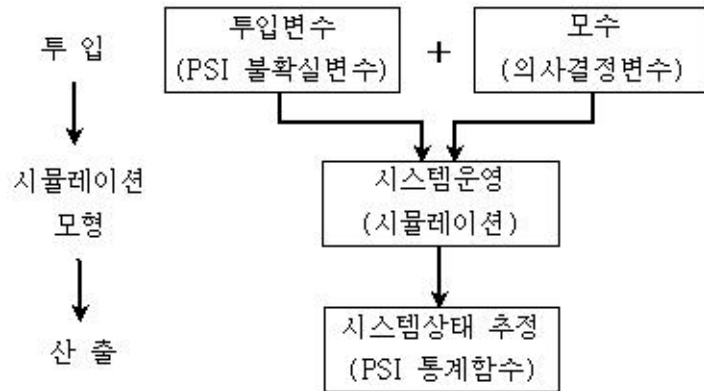
이상에서 살펴 본 대로 스프레드시트에서 시뮬레이션을 통해서 불확실한 상황이 포함된 시스템을 시뮬레이션하여 예측(평가)할 때 기존의 엑셀에서는 대부분의 시간이 수식을 읽고 해석하는 데와 셀에서 데이터를 읽고 계산결과를 스프레드시트 셀에 다시 입력하는 데 시간이 낭비된다. 그러나 PSI 기술을 이용하면 이러한 낭비시간을 없앨 수 있다. 앞에서 살펴본 예제에서는 약 1/10로 줄었으나, PSI기술에 대한 Risk Solver Engine 사용자 가이드를 보면 1/100배까지 줄여서 시간을 절약할 수 있다고 보고하고 있다(User Guide p.40).

III. PSI 기술을 이용한 시뮬레이션과 최적화

1. PSI 기술을 이용한 시뮬레이션

분석하거나 예측하고자 하는 대상 시스템(system)을 시뮬레이션 모형으로 구축하는 과정은 <그림 4>와 같이 "투입 → 시뮬레이션 모형 → 산출" 으로 나타낼 수 있다.

<그림 5> PSI 기술을 이용한 시뮬레이션 구축 과정



앞의 예에서 본다면, 수요량과 판매가격은 투입변수에 해당하고, 순이익은 산출로써 시스템의 상태변수(state variable)에 해당한다. 만일 생산량을 별도의 변수(<그림 4>의 모수에 해당)를 설정한다면, 그것은 통제 가능한 의사결정변수에 해당한다. 의사결정변수의 최적값을 구하는 과정은 "시뮬레이션 최적화"로서 다음 절에서 논의하기로 한다.

PSI 기술을 이용하여 시뮬레이션 모형을 수립하는 과정은 <그림 4>와 같이 3개 모듈로 설명할 수 있다. 첫 번째 모듈은 투입변수(input data)와 함께 불확실변수(uncertain variables)를 입력하는 모듈이다. 이 때 불확실변수는 PSI의 분포함수들을 이용한다. 이와 같은 시뮬레이션 모형에서 투입변수 중 고정된(확정된) 데이터를 제외하고, 변동이 가능한 불확실한 변수(uncertain variables)는 PSI 불확실변수로 나타낸다. 아래 <표 1>은 대표적인 PSI 불확실변수로서 PSI 분포함수(PSI Distributions)라고도 한다.

PSI 분포함수에 추가적인 인수로 PSI 속성함수(PSI property functions)가 사용될 수 있다. PSI 속성함수는 불확실변수들 간의 상관관계설명이나 불확실변수의 이름 정의 등을 위해 사용되며, <표 2>에 제시된 함수들이 있다.

<표 1> PSI 분포함수(Using PSI Distribution Functions)

함수명	함수기능
PsiBinomial(n, p, ...)	n은 시행횟수, p는 성공확률
PsiPoisson(λ , ...)	λ 는 포아송분포의 평균 및 분산
PsiGeometric(p, ...)	p는 성공확률, 첫 번째 성공사건까지의 실패사건 수
PsiHyperGeo(n, D, M, ...)	n개 중 D개가 불량일 때, n번 비복원추출에서 불량개수
PsiIntUniform(a, b, ...)	균등분포를 따르면서 a, b 사이 정수 발생, a와 b는 정수
PsiBeta(α_1 , α_2 , ...)	$\alpha_1=1$, $\alpha_2=1$ 이면 Uniform(0, 1) 생성
PsiBetaGen(α_1 , α_2 , a, b, ...)	[a, b] 사이의 베타분포 생성
PsiBetaSubj(a, c, μ , b, ...)	a: minimum, c: most likely, μ : mean, b: maximum
PsiChiSquare(df, ...)	자유도 df인 카이제곱 분포 생성
PsiExponential(β , ...)	평균이 β 지수함수 생성
PsiGamma(α , β , ...)	$\alpha=1$ 이면 지수분포, $\alpha=df/2$, $\beta=2$ 이면 자유도 df인 카이제곱분포 생성
PsiLogNormal(μ , σ , ...)	평균 μ , 표준편차 σ 인 지수정규분포
PsiNormal(μ , σ , ...)	평균 μ , 표준편차 σ 인 정규분포 생성
PsiPert(a, c, b, ...)	PERT분석의 활동시간변수(a: 최소, c: 평균, b: 최대)
PsiStudent(df, ...)	자유도 df인 t분포변수 생성

<표 2> PSI 속성함수(Using PSI Property Functions)


함수명	함수기능
PsiCorrMatrix(matrix cell range, position, instance)	하나의 불확실변수와 다른 불확실변수그룹 사이의 상관관계 설명
PsiCorrDepen(corrname, coefficient)	하나의 불확실변수와 다른 불확실변수 사이의 상관관계 설명
PsiCorrIndep(corrname)	독립변수인 불확실변수와 종속변수인 다른 불확실변수와의 상관관계 설명
PsiLock(value)	불확실변수를 일시적으로 고정
PsiName(name)	불확실변수에 이름 지정
PsiShift(shift)	일정량 불확실변수의 분포 영역을 이동
PsiTruncate(min, max)	최소~최대 사이에 있는 불확실변수의 표본값 제한

두 번째 모듈은 입력된 변수를 가지고 시스템의 작동과정을 모방하는 모듈이다. 이 모듈에서는 대상 시스템의 작동과정을 엑셀 스프레드시트 모형으로 표현한다.

세 번째 모듈은 시뮬레이션 결과를 PSI 통계함수로 도출하는 모듈이다. PSI 통계함수는 PSI 불확실함수(uncertain function)로도 불리며 관련된 주요한 함수들은 <표 3>과 같다.

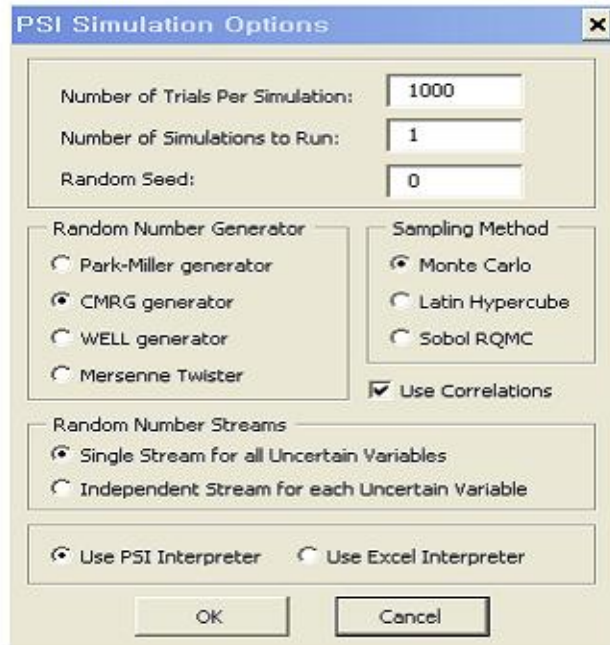
<표 3> PSI 통계함수(Using PSI Statistics Functions)

함 수 명	함 수 기 능
PsiMean()	지정된 셀의 평균값 산출
PsiVariance()	지정된 셀의 분산 산출
PsiStdDev()	지정된 셀의 표준편차 산출
PsiPercentile(cell, percentile, simulation)	지정된 셀의 퍼센타일 산출
PsiBVaR()	지정된 셀의 위험가치(VaR)를 측정
PsiCVaR()	지정된 셀의 조건적인 위험가치(VaR)를 측정

이 세가지 모듈을 구축하여 시뮬레이션 모형을 수립한 다음 PSI 실시간 시뮬레이션 버튼(Enable Interactive Simulation 버튼, 엑셀 시트상의 메뉴바에 '전구(☉)' 모양의 돌바 버튼)을 클릭하여 시뮬레이션을 실행하면, PSI 통계함수에는 시뮬레이션 결과가 나타난다. 여기서 돌바의 PSI 도구버튼(PSI Simulation Option, )을 누르면 시뮬레이션 수와 실행횟수, 난수산출방법 등 여러 가지 옵션을 선택할 수 있다.

대부분의 시뮬레이션은 <그림 7>의 첫 번째 그룹에 있는 옵션을 주로 설정한다. 현재 <그림 7>은 한 번 시뮬레이션을 1,000번 반복하도록 디폴트값이 정해져 있으며, Random Seed는 0으로 설정되어 있다. 여기서 Random Seed 값이 0으로 설정되면 매 시뮬레이션마다 다른 난수가 생성됨을 의미한다. 만일 다수의 시뮬레이션을 실행하고자 한다면 Number of Simulations to Run을 변화시키고, 각 시뮬레이션별로 시행횟수 증대시키고자 한다면 Number of Trials Per Simulation을 증가시키면 된다. 또한 하나의 시뮬레이션과 다른 시뮬레이션의 결과 값이 유사하길 원한다면, Random Seed에 0이 아닌 값을 입력하여 각 시뮬레이션에 같은 난수를 사용할 수 있다.

<그림 8> PSI Simulation Option 설정



보통의 시뮬레이션에서는 일반적인 난수 생성만으로 충분하다. 그러나 다양한 불확실변수를 포함하는 금융공학과 같은 시뮬레이션에서는 좀 더 다양한 난수 생성을 필요로 한다. 이러한 용도에 따라 Risk Solver Engine은 <그림 7>의 Random Number Generator에서 4가지 난수생성방법 중 하나를 선택하도록 한다. Park-Miller Generator는 주기가 $2^{31}-2$, Combined Multiple Recursive Generator(CMRG)는 주기가 2^{191} , Well Equidistributed Long-period Linear(WELL) generator는 주기가 2^{1024} , Mersenne Twister는 주기가 $2^{19937}-1$ 인 난수를 생성한다. 여기서 주기가 길면 길수록 같은 난수가 발생하지 않으며 난수가 독립적으로 산출된다. 이러한 4가지 난수생성방법 중 시뮬레이션에 더 유용한 방법을 택하면 된다.

대부분의 Monte Carlo 시뮬레이션 도구는 0과 1사이에서 단일 시퀀스(스트림)의 난수를 생성한다. 이 경우 생성된 난수들 사이에 매우 미묘한 의존관계가 있을 수 있으나(한쪽으로 치우침이 발생할 수도 있음), 일반적 시뮬레이션에서는 크게 영향을 미치지 않는다. 그러나 금융공학에 활용되는 시뮬레이션은 더 나은 결과를 위해 난수의 시퀀스가 독립성을 유지할

필요가 있다. 따라서 Risk Solver Engine은 Single Stream for all Uncertain Variables와 Independent Stream for each Uncertain Variables의 2가지 옵션을 제공하고 있다.

또한 Risk Solver Engine은 표본추출방법으로 Monte Carlo sampling외에 0과 1사이를 소구간으로 나누고 소구간에서 균등하게 난수를 산출하는 Latin Hypercube와 주로 계량적 재무문제의 시뮬레이션에 적용하는 Sobol RQMC(Randomized QMC)를 추가로 제공하고 있다. 만일 난수들이 상관관계를 가지면서 랜덤하게 발생해야 할 때는 Use Correlations를 체크하면 된다.

그리고 마지막으로 Risk Solver Engine은 시뮬레이션에서 PSI 기술을 적용할 것인지 아니면 엑셀의 방법을 적용할 것인지 선택할 수 있게 한다. Use PSI Interpreter를 선택할 경우 기존 엑셀에서 시뮬레이션 할 때 보다 약 100배 정도 더 빠르게 계산한다. 그러나 PSI 해석기가 사용될 수 없는 경우(해당 메시지가 뜬) Use Excel Interpreter를 체크한다.

2. PSI 기술을 이용한 최적화

시뮬레이션에는 시스템 설계 또는 운영에 관계되는 의사결정변수가 관여되기 마련이다. <그림 4>에서 보면 오른쪽 위의 모수(parameter)에 해당하는 부분으로 그 중에서 통제가능한 부분이 의사결정변수가 된다. 보통은 의사결정변수의 값을 임의로 정한 다음, 그 값의 경우에 한정하여 시뮬레이션을 수행한 후, 시스템의 상태값(성과 특정치)을 측정한다. 의사결정변수에 여러 가지 다양한 값을 시도하여 그 중 우수한 값(성과특정치)을 주는 의사결정변수 값을 최적의 해를 선정하여 시스템을 설계하거나 운영한다. 최근에 와서는 이러한 수동적인 과정을 시뮬레이션과 연계하여 자동으로 수행하는 시뮬레이션 최적화 기능을 갖춘 패키지들이 있다. 예를 들면, @Risk에서는 @Risk Evlove와 @RiskOptimizer라는 프로그램을 공급하고, CrystalBall은 QptQuest⁵⁾라는 프로그램을, Arena에서도 OptQuest를 제공하고 있다.

이러한 프로그램들은 시뮬레이션 최적화에 사용하기가 불편하였다. 최적화에 포함될 의사결정변수와 제한조건을 엑셀환경이 아닌 별도의 환경에서 입력하고 모형화하여야 하기 때문이다. 따라서 엑셀의 해찾기(Solver) 기능에 익숙한 일반 사용자에게는 엑셀 해찾기(Solver) 기능을 엑셀의 시뮬레이션 모형에 직접 적용하여 사용할 수 있기를 희망하였다. 이러한 수요에 부응하여 Frontline Systems 사는 PSI 기술을 개발하였다. 즉 PSI 기술을 이용한 시뮬레이션 모형을 수립하면, 거기서 바로 엑셀의 해찾기(Solver) 기능을 적용할 수 있다.

5) <http://www.opttek.com/>의 Scatter Search 및 Tabu Search 를 이용한 최적화 프로그램.

Frontline Systems사의 Premium Solver Platform라는 업그레이드된 Solver를 설치하면 PSI를 이용한 시뮬레이션의 최적해를 구할 수 있다.

다음 예제는 한정된 예산으로 다양한 프로젝트 대안에 대한 투자안을 선택하는 자본예산에 대한 시뮬레이션 최적화 문제이다. 성공과 실패에 대한 확률이 주어지며 성공시 현금 흐름에 대한 확률이 주어진 8가지 프로젝트를 가정한다. 알려진 정보는 올해에 각 프로젝트에 대한 투자금액이다(<그림 8>의 Initial Investment 참조). 8개 프로젝트 모두에 투자한다면 초기 투자금액은 총 \$2,500,000이다. 현재 투자 가능한 예산은 \$1,500,000이다. 따라서 주어진 예산에 맞추어 최대의 기대 수익을 낼 수 있도록 8가지 프로젝트 중 일부를 선택해만 한다.⁶⁾

<그림 9> Premium Solver를 이용한 PSI 시뮬레이션 최적화 예

Project Number	Cash Flow If Successful	Chance of Success	Expected Cash Flow	Initial Investment	Expected Cash -	Decision to Include
1	\$794,740	100%	\$794,740	\$325,000	\$469,740	1
2	\$671,763	100%	\$671,763	\$450,000	\$221,763	1
3	\$574,238	100%	\$574,238	\$550,000	\$24,238	1
4	\$615,502	100%	\$615,502	\$300,000	\$315,502	1
5	\$519,515	100%	\$519,515	\$150,000	\$369,515	1
6	\$541,436	100%	\$541,436	\$250,000	\$291,436	1
7	\$459,541	100%	\$459,541	\$150,000	\$309,541	1
8	\$490,788	100%	\$490,788	\$325,000	\$165,788	1
Total Investment				2,500,000		
Budget Constraint				1,500,000		
Total Cash Flow				2,167,522		
Expected Total CF				1,983,358		

성공시 현금흐름에 대한 확률을 모델링하기 위하여 Cash Flow If Successful 열(C열)에 PsiTriangular() 함수를 사용한다. 예를 들어 셀 C4 =PsiTriangular(400000,500000,900000)

6) 본 예는 Premium Solver Platform의 free trial에 포함된 예제임.

가 입력된다. 또한, 성공확률을 모델링하기 위해 Chance of Success열(D열)에 =PsiBinomial(1,0,9)를 입력하여 준다. 이 이항분포는 매 시행에서 90% 확률로 1의 값, 10% 확률로 0의 값이 되도록 해준다. 그런 다음 미래에 기대할 수 있는 현금흐름을 계산하기 위해 C열과 D열을 곱한 값을 Expected Cash Flow(E열)에 입력하고, 다시 Expected Cash Flow에서 Initial Investment(F열)를 차감한 값을 Expected Cash(G열)에 입력하여 미래의 예상수익을 계산한다. Decision to Include(H열)은 프로젝트를 선택하면 1, 그렇지 않으면 0으로 입력한다. 마지막으로 총 현금흐름을 나타내는 셀 F16의 Total Cash Flow는 SUMPRODUCT 함수를 활용한다. 이는 각 프로젝트별 G열과 H열의 값들을 곱해서 모두 더해주는 것으로, 셀 F16 =SUMPRODUCT(G4:G11,H4:H11)+PsiOutput()가 된다. 그리고 셀 F17에는 =PsiMean(F16)을 입력하여 기대한 현금흐름 총액의 평균을 산출한다.

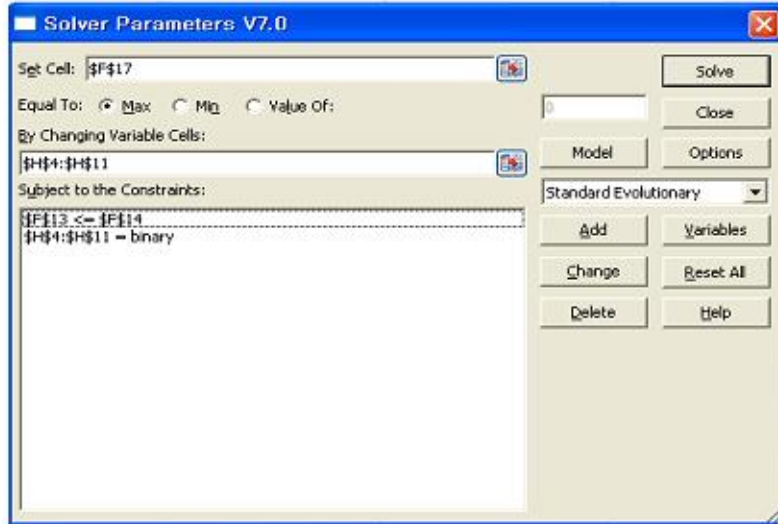
이상과 같이 모형화 한 다음 전구모양의 아이콘이나 F9를 클릭하면 이러한 계산을 1,000번을 반복시킬 수 있으며 매 1,000번의 시뮬레이션마다 평균값(F16)이 변화되는 것을 관찰할 수 있다.

이 예에서 최종 목적은 기대 현금의 총액인 F16의 Total Cash Flow를 최대화하는 것이다. 그리고 의사결정변수는 Decision to Include가 되며 프로젝트의 선택은 1, 거절은 0이 되도록 하는 binary(이진수)를 적용한다. 셀 F13은 프로젝트가 선택될 경우 그 선택된 프로젝트의 초기 투자 총액으로 =SUMPRODUCT(F4:F11,H4:H11)가 되며 셀 F14의 주어진 예산 \$1,500,000보다 작거나 같아야 한다. 따라서 도구의 Premium Solver를 선택하여 다음 <그림 9>와 같이 입력한 후 Solve를 클릭한다.

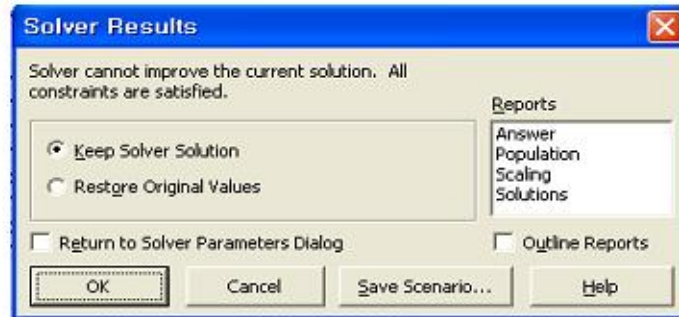
Solve를 클릭하면 해 찾기가 실행된다. 몇 초의 시간이 지나면 해 찾기가 끝나고 최적해를 구해준다. 다음 <그림 10>과 같은 결과값에 대한 다양한 옵션을 선택할 수 있다. 여기서 필요한 Reports의 항목을 클릭하고 완료된 해 찾기 값으로 전환하기 위해 Keep Solver Solution을 체크한 후 OK를 클릭한다.

최종적으로 도출된 시뮬레이션 결과는 <그림 11>에 보는 것처럼 주어진 예산내에서 프로젝트 1, 4, 5, 7의 선택하게 되고 이에 따른 기대한 현금흐름의 총액은 \$621,689가 된다. 이러한 결과값에 대한 좀 더 유용한 정보들은 앞서 <그림 10>의 Solver Results의 Reports 중 Answer나 Population을 선택하여 자세히 살펴 볼 수 있으며 그 값은 아래 <그림 12>과 <그림 13>과 같다.

<그림 10> Premium Solver를 이용한 PSI 시뮬레이션 최적화



<그림 11> PSI를 이용한 최적화 Solver Results



<그림 12>의 Answer Report를 보면 본 시뮬레이션은 진화알고리즘(유전자 알고리즘)을 사용했고, 총 시뮬레이션에 사용된 시간이 1분 4초 이며, 2,672개의 하위문제를 해결하였음을 알 수 있다. 또한 시뮬레이션의 목표셀과 변경셀, 제약식이 요약되어 있으며 해당 최종 결과값을 나타내어 준다.

PSI 기술을 이용한 시뮬레이션과 최적화

<그림 12> PSI를 이용한 최적화의 결과

Project Number	Cash Flow if Successful	Chance of Success	Expected Cash Flow	Initial Investment	Expected Cash -	Decision to Include
1	\$611,878	100%	\$611,878	\$325,000	\$286,878	1
2	\$585,873	100%	\$585,873	\$450,000	\$135,873	0
3	\$1,227,137	100%	\$1,227,137	\$550,000	\$677,137	0
4	\$560,901	100%	\$560,901	\$300,000	\$260,901	1
5	\$369,620	100%	\$369,620	\$150,000	\$219,620	1
6	\$496,043	100%	\$496,043	\$250,000	\$246,043	0
7	\$497,153	100%	\$497,153	\$150,000	\$347,153	1
8	\$501,499	100%	\$501,499	\$325,000	\$176,499	0
Total Investment				925,000		
Budget Constraint				1,500,000		
Total Cash Flow				1,114,552		
Expected Total CF				621,689		

<그림 13> PSI를 이용한 최적화 Answer Report

Microsoft Excel 10.0 Answer Report

Report Created: 11/29/2006 01:21:59 오전

Result: Solver cannot improve the current solution. All constraints are satisfied.

Engine: Standard Evolutionary

Solution Time: 01 Minutes, 04 Seconds

Iterations: 0

Subproblems: 2672

Incumbent Solutions: 2949

Cell	Name	Original Value	Final Value
\$F\$17	Expected Total CF Initial Investment	598968.1073	621689.0086

Cell	Name	Original Value	Final Value
\$H\$4	Decision to Include	1	1
\$H\$5	Decision to Include	0	0
\$H\$6	Decision to Include	1	0
\$H\$7	Decision to Include	1	1
\$H\$8	Decision to Include	1	1
\$H\$9	Decision to Include	0	0
\$H\$10	Decision to Include	1	1
\$H\$11	Decision to Include	0	0

Cell	Name	Cell Value	Formula	Status	Slack
\$F\$13	Total Investment Initial Investment	925,000	=\$F\$13<= \$F\$14	Not Binding	1175000

또한 <그림 13>은 본 시뮬레이션의 Population Report이다. 진화 알고리즘은 한 번에 생성할 Population의 크기와 반복횟수를 정하며 정해진 크기에 따라 Population을 랜덤하게 생성한다. 반복된 시뮬레이션 결과 좋은 Population원소만을 남게 되는데 Population Report는 진화 알고리즘에서 선택된 좋은 Population에 대한 정보(최적해, 평균, 표준편차 등)를 요약하여 보여준다.

<그림 14> PSI를 이용한 최적화 Population Report

The screenshot shows an Excel spreadsheet titled 'Microsoft Excel 10.0 Population Report'. The main content is a table with two sections: 'Adjustable Cells' and 'Constraints'.

Cell	Name	Best Value	Mean Value	Standard Deviation	Maximum Value	Minimum Value
\$H\$4	Decision to Include	1	0.4375	0.499208234	1	0
\$H\$5	Decision to Include	0	0.375	0.487177352	1	0
\$H\$6	Decision to Include	0	0.175	0.382364396	1	0
\$H\$7	Decision to Include	1	0.4625	0.501737487	1	0
\$H\$8	Decision to Include	1	0.3875	0.490253099	1	0
\$H\$9	Decision to Include	0	0.1	0.301892763	1	0
\$H\$10	Decision to Include	1	0.3625	0.48375509	1	0
\$H\$11	Decision to Include	0	0.35	0.479978902	1	0

Cell	Name	Best Value	Mean Value	Standard Deviation	Maximum Value	Minimum Value
\$F\$13	Total Investment Initial Investment	925,000	797,188	387420.7504	1475000	0

참고로 Frontline System社의 최적화 프로그램을 소개하면 다음과 같다.

첫 번째, Solver(해찾기)는 마이크로소프트 엑셀에 번들로서 공급되고 있는데, 선형, 비선형 정수계획모형의 문제를 풀 수 있다.

두 번째, Premium Solver는 non-smooth problem(NSP)은 evolutionary solver(유전자 해법적용)을 적용하고, smooth nonlinear problem(NLP)에는 GRG solver를 적용해서, 변수 400개, 제약식 200개 까지 풀 수 있다. 선형계획 문제는 변수 2,000개, 제약식 1,000개 까지 처리할 수 있다. Evolutionary solver는 엑셀에서 IF, CHOOSE, LOOKUP함수등으로 표현된 non-smooth 함수를 처리할 수 있다. 또 정수 계획에서도 all different 유형의 제약식 문제도 풀 수 있다. 이는 TPS(Traveling Salesman Problem)과 같이 순열제약의 문제를 풀 수

있다. 버전 V7.0에서는 엑셀 Excel 2007에서 16,384개 행의 워크시트를 처리할 수 있으며, Risk Solver Engine의 시뮬레이션 최적화 문제를 풀 수 있다.

세 번째, Premium solver Platform V7.0은 Frontline System사의 간판격인 소프트웨어이다. 여기에는 LP/Quadratic solver(선형 및 2차식 모형문제)는 8,000개 변수까지의 문제를 풀 수 있다. 정수형 문제에 대해서도 기존의 solver에 비해서 1,000배의 빠른 속도로 문제를 푼다. 이 외에도 nonlinear GRG Solve, Evolutionary Solver, Interval Global solver, SOCP Barrier solver등을 제공한다. 이들에 대한 자세한 내용은 Risk Solver Engine Use Guide pp.14-18에서 볼 수 있다. 무엇보다도 특징적인 것은 PSI 기술을 더 큰 대형사이즈의 문제(2,400만개 변수까지)를 더 빠르게 풀 수 있다는 것이다.

네 번째, Premium Solver(교육용)는 대학의 학부와 경영대학원(MBA) 코스에서 채택하는 경영과학의 교과서에 교육용으로 제공하는 것으로 Premium Solver Platform과 동일한 성능(속도와 문제사이즈에 대한 것은 제외)을 갖고 있다. 200개 변수와 제약식 100개 까지를 처리할 수 있다.

앞서 이야기 했듯이 시뮬레이션 최적화를 위해 많은 소프트웨어 회사들이 다양한 기법들을 제공하고 있다. 다음 <표 4>는 그 중에서 엑셀의 해찾기(Solver)와 Premium Solver Platform, @Risk의 Evolutionary Search(Evolver)를 비교한 것이다.⁷⁾

7) <http://www.palisade.com.au/> 및 Premium Solver Platform User Guide를 재구성

<표 4> 엑셀 Solver, @Risk Evlover, Premium Solver Platform의 비교

	Excel Solver	@Risk의 Evlover	Premium Solver Platform
문제종류	linear simple non-linear	linear / nonlinear combinatorial(조합) order/permutation(순열) scheduling(일정계획) grouping mixed types customised	linear nonlinear quadratic quadratically constrained second order cone ⋮
방법론	Simplex Method or Branch & Bound Method (linear) GRG2(non-linear)	Genetic Algorithm	Evolutionary (Genetic Algorithm) IF, CHOOSE, LOOK UP 같은 함수처리 가능 GRG(non-smooth 함수는 곤란) Branch & Bound Method
변수	200	80-unlimited	400-8,000

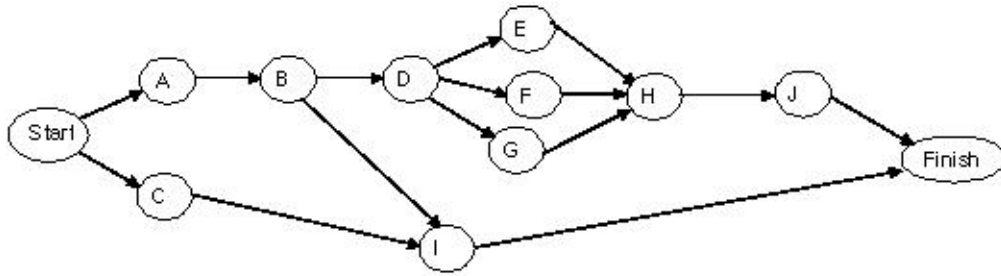
엑셀의 해찾기(Solver)가 제한적 문제에 제한적인 방법론을 제시하는 반면 @Risk Evlover와 Premium Solver Platform은 좀 더 다양한 문제에 보다 나은 방법론을 제시하고 있다. 그러나 @Risk Evlover의 경우 방법론이 유전자 알고리즘 하나인 반면 Premium Solver Platform은 다양한 방법론을 제시하고 있으며, 더 나아가 <표 4>에 제시된 것 이외에 Large-Scale의 문제를 해결할 수 있도록 하고 있다. 자세한 내용은 User Guide를 참조하기 바란다.

IV. PSI 기술을 이용한 시뮬레이션의 유의할 점

PSI 기술을 이용하여 시뮬레이션을 할 때, 불확실변수(uncertain variables)를 수차례 더하거나 뺀 결과를 가지고 그것이 0인지 검사하는데서 오류가 발생하는 것을 발견하였다. 이것을 예시하기 위해서 PERT 네트워크를 가지고 @Risk 시뮬레이션 결과와 PSI 시뮬레이션 결과

를 비교해 보았다. 이 예제는 Albright(2005)의 예제(p. 507)를 편의상 그대로 사용하였다.

<그림 15> 네트워크 경로 및 각 활동시간의 PERT 분포



Activities	a	b	c
A	1.5	3.5	8.5
B	3	4	5
C	7	10	19
D	2	2.5	6
E	3	3.5	7
F	2	2.5	6
G	2	4	6
H	2.5	3	3.5
I	0.5	1	1.5
J	1.5	2	2.5

각 활동의 흐름은 위의 네트워크와 같으며 각 작업에 대한 시간은 [a, c] 사이의 PERT 분포를 따른다고 가정하였다. 위 데이터를 바탕으로 어느 활동이 주요활동인지 알아보기 위해서 우선 @Risk를 이용하여 1000번의 시뮬레이션을 실행하였다.

F6 셀에 =RiskPert(B6, C6, D6)를 입력하고 F15셀까지 복사함으로써 PERT 분포에 의한 각 활동의 소요시간을 발생시켰다. 그리고 IF 함수를 이용하여 여유시간이 0일 때를 1로 표현하고, 여유시간이 0보다 크면 0으로 표현하였다. 즉, G21 셀에 =IF(\$F21=0, 1, 0) 함수를 사용하고, H21에는 =RiskMean(G21)를 사용하였다.

F6에는 =RiskPert(B6, C6, D6), F7:F15까지 복사하여 사용.

F21에는 = E21-B21, F22:F30 까지 복사하여 사용.

G21에는 = IF (F21 = 0, 1, 0), G22:GF30 까지 복사하여 사용.

H21에는 =RiskMean(G21), H22:H30 까지 복사하여 사용.

<그림 16> @Risk를 이용한 PERT 네트워크 시뮬레이션 결과

	A	B	C	D	E	F	G	H
1								
2								
3	Parameters of PERT distributions							
4	Label	Min	Most likely	Max	Implied mean	Duration		
5	Start					0		
6	A	1.5	3.5	8.5	4	4.000		
7	B	3	4	5	4	4.000		
8	C	7	10	19	11	11.000		
9	D	2	2.5	6	3	3.000		
10	E	3	3.5	7	4	4.000		
11	F	2	2.5	6	3	3.000		
12	G	2	4	6	4	4.000		
13	H	2.5	3	3.5	3	3.000		
14	I	0.5	1	1.5	1	1.000		
15	J	1.5	2	2.5	2	2.000		
16	Finish					0		
17								
18	Activity start and finish times							
19	Activity	Earliest start time	Latest finish time	Latest finish time	Earliest start time	Slack	critical path?	Prob(critical)
20	Start	0.00	0.00	0.00	0.00	0.00		
21	A	0.00	4.00	4.00	0.00	0.00	1	0.963
22	B	4.00	8.00	8.00	4.00	0.00	1	0.963
23	C	0.00	11.00	19.00	8.00	8.00	0	0.001
24	D	8.00	11.00	11.00	8.00	0.00	1	0.999
25	E	11.00	15.00	15.00	11.00	0.00	1	0.447
26	F	11.00	14.00	15.00	12.00	1.00	0	0.071
27	G	11.00	15.00	15.00	11.00	0.00	1	0.481
28	H	15.00	18.00	18.00	15.00	0.00	1	0.999
29	I	11.00	12.00	20.00	19.00	8.00	0	0.001
30	J	18.00	20.00	20.00	18.00	0.00	1	0.999
31	Finish	20.00	20.00	20.00	20.00			

어떤 활동의 빠른 시작시간(Earliest Start Time)과 늦은 시작시간(Latest Start Time) 값의 차인 여유시간(Slack Time)이 0이면 그 활동은 주요활동(critical activities)이 된다. 1000번의 시뮬레이션 동안, G21:G30은 각 활동이 주요활동이 되면 1이 되고, 그것의 평균(RiskMean)인 H21:H30은 각 활동이 주요활동이 될 확률 된다. 각 활동이 주요활동이 될 확률은 그림의 H21:H30과 같이 분포하였다. 즉 A, B, D, H, J 활동이 주요활동임을 알 수 있다. 한 가지 언급할 것은 각 활동이 주요활동이 될 확률을 표현할 수 있는 것은 PERT 분석에 시뮬레이션을 적용한 효과로서 시뮬레이션이 아니면 불가능한 것이다.

이것을 PSI 기술을 이용하여 시뮬레이션 해보았다. <그림 16>은 PSI 기술을 이용한 시뮬레이션 결과이다(디플트로 1000번의 반복이 설정).

<그림 17> PSI를 이용한 PERT 네트워크 시뮬레이션 결과(PSI 해석자 사용)

	A	B	C	D	E	F	G	H
1								
2								
3	Parameters of PERT distributions							
4	Label	Min	Most likely	Max	Implied mean	Duration		
5	Start					0		
6	A	1.5	3.5	8.5	4	3.227		
7	B	3	4	5	4	4.525		
8	C	7	10	19	11	15.359		
9	D	2	2.5	6	3	2.049		
10	E	3	3.5	7	4	5.300		
11	F	2	2.5	6	3	3.928		
12	G	2	4	6	4	5.108		
13	H	2.5	3	3.5	3	3.268		
14	I	0.5	1	1.5	1	1.048		
15	J	1.5	2	2.5	2	2.066		
16	Finish					0		
17								
18	Activity start and finish times							
19	Activity	Earliest start time	Earliest finish time	Latest finish time	Earliest start time	Slack	critical path?	Prob(critical)
20	Start	0.00	0.00	0.00	0.00			
21	A	0.00	3.23	3.23	0.00	0.00	1	0.171
22	B	3.23	7.75	7.75	3.23	0.00	1	0.171
23	C	0.00	15.36	19.39	4.03	4.03	0	0.001
24	D	7.75	9.80	9.80	7.75	0.00	1	0.374
25	E	9.80	15.10	15.10	9.80	0.00	1	0.216
26	F	9.80	13.73	15.10	11.17	1.37	0	0.035
27	G	9.80	14.91	15.10	9.99	0.19	0	0.236
28	H	15.10	18.37	18.37	15.10	0.00	1	0.67
29	I	15.36	16.41	20.44	19.39	4.03	0	0.001
30	J	18.37	20.44	20.44	18.37	0.00	1	0.942
31	Finish	20.44	20.44	20.44	20.44			

앞에서 입력한 입력수식(함수)을 그대로 사용하되, 다음 두 가지만 수정하였다.

F6 = PsiPert(B6, O6, D6), F15까지 복사하여 사용.

G21 = IF(F21 = 0, 1, 0) G30 까지 복사하여 사용(동일함).

H21 = PsiMean(G21), H30까지 복사하여 사용.

PSI를 이용한 시뮬레이션 결과 J활동만 1에 가까운 주요활동으로 나타났으며, 다른 주요 활동은 구분해내지 못하였다. 이 결과는 @Risk를 이용한 시뮬레이션 결과와 크게 차이를 보여주었다. 결과를 보면 PERT 네트워크에서 PSI기술은 주요활동은 구분해 내지 못하고 있다(H21:H30을 참조). PSI함수를 이용하되, 옵션에서 PSI 해석자가 아닌 엑셀 해석자(Excel Interpreter 선택)를 이용하여 시뮬레이션을 해보았다(PSI 시뮬레이션 옵션에서 설정할 수 있음. 앞의 Ⅲ절 <그림 7> 참조). 그 결과는 다음의 <그림 17>과 같다.

<그림 18> PSI를 이용한 PERT 네트워크 시뮬레이션 결과(엑셀 해석자 사용)

	A	B	C	D	E	F	G	H	
1									
2									
3		Parameters of PERT distributions							
4	Label	Min	Most likely	Max	Implied mean	Duration			
5	Start					0			
6	A	1.5	3.5	8.5	4	4.218			
7	B	3	4	5	4	3.653			
8	C	7	10	19	11	13.139			
9	D	2	2.5	6	3	2.980			
10	E	3	3.5	7	4	5.312			
11	F	2	2.5	6	3	2.722			
12	G	2	4	6	4	3.714			
13	H	2.5	3	3.5	3	3.201			
14	I	0.5	1	1.5	1	0.994			
15	J	1.5	2	2.5	2	1.733			
16	Finish					0			
17									
18	Activity start and finish times								
19	Activity	Earliest start time	Earliest finish time	Latest finish time	Earliest start time	Slack	critical path?	Prob(critical)	
20	Start	0.00	0.00	0.00	0.00	0.00			
21	A	0.00	4.22	4.22	0.00	0.00	1	0.959	
22	B	4.22	7.87	7.87	4.22	0.00	1	0.959	
23	C	0.00	13.14	20.10	6.96	6.96	0	0	
24	D	7.87	10.85	10.85	7.87	0.00	1	0.999	
25	E	10.85	16.16	16.16	10.85	0.00	1	0.453	
26	F	10.85	13.57	16.16	13.44	2.59	0	0.054	
27	G	10.85	14.56	16.16	12.45	1.60	0	0.493	
28	H	16.16	19.36	19.36	16.16	0.00	1	1	
29	I	13.14	14.13	21.10	20.10	6.96	0	0	
30	J	19.36	21.10	21.10	19.36	0.00	1	1	
31	Finish	21.10	21.10	21.10	21.10				

엑셀 해석자를 이용한 PSI 시뮬레이션 결과는 활동 A, B, D, H, J가 1에 매우 근접한 주요활동으로 나타났다. 이는 @Risk를 이용한 시뮬레이션 결과와 동일하다.

이는 PSI 해석기가 불확실변수의 몇 개 더해지거나 빠질 때 에러가 누적되어서 그 결과가 0인지 구분하지 못하기 때문인 것으로 판명되었다. 참고로, G21의 IF함수를 다음과 같이 절대값을 이용하여 에러를 허용하면 올바른 결과를 얻을 수 있었다.

G21 = IF(ABS(F21) <= 0.0001, 1, 0), G30 까지 복사하여 사용.

위 시뮬레이션 결과들을 모두 요약하면 다음의 <표 5>와 같다.

<표 5> PERT 네트워크 시뮬레이션 결과 요약

활동	@Risk를 이용한 시뮬레이션	PSI를 이용한 시뮬레이션(PSI해석자)	PSI를 이용한 시뮬레이션 (엑셀해석자)
A	0.963	0.171	0.959
B	0.963	0.171	0.959
C	0.001	0.001	0
D	0.999	0.374	0.999
E	0.447	0.216	0.453
F	0.071	0.035	0.054
G	0.481	0.236	0.493
H	0.999	0.67	1
I	0.001	0.001	0
J	0.999	0.942	1
주요활동	A, B, D, H, J	J	A, B, D, H, J

@Risk와 PSI는 모두 엑셀 스프레드시트 상에서 시뮬레이션을 용이하게 해주는 도구이지만 그 결과는 다른 양상을 보였다. 엑셀 해석자를 이용한 PSI 시뮬레이션의 결과는 @Risk 시뮬레이션 결과와 동일하게 나왔지만, PSI 해석자를 이용한 PSI 시뮬레이션 결과는 @Risk와 다른 결과를 보였다. 따라서 PSI 기술을 이용해서 시뮬레이션을 할 때, IF 함수로서 "0"인지 체크하는 경우, 그것이 PSI 불확실함수를 포함하고 있으면 반드시 엑셀 해석자를 이용하여 시뮬레이션을 실행해야 한다.

V. 결 론

본 논문에서는 PSI 기술을 이용한 엑셀 시뮬레이션을 소개하였다. 간단한 예제를 통해서 시뮬레이션(Monte Carlo Simulation)이 필요함을 보였으며, 엑셀에서 시뮬레이션 모형을 직접 수립하는 경우와 PSI 기술을 이용하여 시뮬레이션 모형을 수립하는 것을 보였다. 시뮬레이션을 비교했을 때 그 소요 시간이 약 1/10로 줄어든 것을 확인하였다.

한편, PSI 기술을 이용한 시뮬레이션 모형을 해찾기(Solver)에 직접 적용한 예시를 보였다. Frontline Systems사의 해찾기 프로그램들을 소개하였다. PSI 기술을 이용해서 시뮬레

이션 하는데 한 가지 주의할 것은 불확실변수를 포함한 셀은 IF 함수로서 0인지 체크할 때에는 반드시 계산상 에러를 고려하여 절대치함수(ABS())로 대체하거나 옵션에서 엑셀 해석자를 이용해야만 정확한 결론을 얻을 수 있다.

참고문헌

- 1) Albright, A. C., W. L. Winston, *Spreadsheet Modeling and Applications: Essentials of Practical Management Science*, Thomson, 2005.
- 2) *Risk Solver Engine User Guide* (2006)
- 3) *Premium Solver Platform User Guide* (2006)
- 4) Grossman, T., "Spreadsheet ADD-INS for OK/MS", *ORMS Today*, August 2002, pp.46-51

참고사이트

- 1) www.solver.com
- 2) <http://www.palisade.com>
- 3) <http://www.decisioneering.com>
- 4) <http://www.opttek.com/>