

# 在庫管理 시뮬레이션\*

## A Study on Inventory System Simulation

李 揆 相\*\*

《 目 次 》

I. 序 論	1. 事象과 變數와 副프로그램
II. 亂數生成과 몬테칼로法	2. 主프로그램
1. 問題의 概要	3. 副프로그램들
2. 亂數生成	IV. 結論: 結果와 分析
3. 몬테칼로法	< 컴퓨터 프로그램 리스트 >
III. 포트란프로그램	< 참 고 문 헌 >

### I. 序 論

計量的 意思決定分野에서 在庫管理問題는 가장 먼저 개발되기 시작하였다. 과학적 在 庫管理를 위하여 數理的 模型을 구성하고, 이 모델에 따라 最適在庫政策을 決定하며, 이제 는 컴퓨터를 많이 사용하여 在庫水準을 기록 유지하며 언제 얼마큼 보충할 것인가를 알아

\* 1) 이 論文은 문교부의 1981년도 經營學 教授 國費 海外 派遣 研究計劃의 지원으로 이루어진 것임.  
 2) 이 論文은 필자가 연구중이던 기관의 Averil M. Law 교수 ( Dept. of Industrial Engineering, Univ. of Wis./Madison, 현재는 Univ. of Ariz./Tucson 교수) 의 프로그래머와 Herman F. Karreman 교수 ( Graduate School of Business Administration, Univ. of Wis./Madison ) 의 확률모델 강의에 많이 힘입었다.

\*\* 牧園大學 經營學科 助教授·財務管理

내고 있다.

그런데 在庫管理分野에서 需要사이의 시간간격, 需要量, 注文調達期間, 이상 세가지 모두가 確率變數를 보인다면 數理的모델로서는 最適解를 구할 수 없고 시뮬레이션에 의한 模型實驗밖에는 별다른 도리가 없다.

시뮬레이션은 반드시 最適解를 주지 못하는 短點등을 갖고 있지만, 數理的 모델보다는 이해하기 쉽고 시행착오의 과오를 감소시킬 수 있는 장점을 갖고 있다.

여기에서는 이러한 시뮬레이션의 장점을 살려 數理的으로 풀 수 없는 在庫管理의 한 분야를 골라 模擬實驗하는 데 目的을 두고 있다. 따라서 어떤 假定을 樹立하고 이를 證明하거나, 또는 數理的 解를 導出해 내는 데에 目的이 있는 것이 아니라 實驗을 設討하고 그 結果가 어떻게 나오는가를 관찰하고 분석하는 것이 目的이다.

위에서 언급한 세가지 모두가 確率變數의 성격을 띠는 (s, S) 在庫模型을 가정하고, 먼저 이들 세 變數에 관하여 컴퓨터 프로그램으로 亂數를 生成시킨뒤, 이들 亂數를 갖고 몬테칼로法으로 實驗을 시행해 보고자 한다.

다음 위의 原理를 이용하여 컴퓨터 프로그램을 만들코자 하는데 이는 FORTRAN에 의하고자 한다.

實驗結果와 分析이 結論에 해당된다고 하겠으며 最適在庫政策의 選擇과 實驗의 長短點등을 살펴보고자 한다. 아래에서는 시뮬레이션을 模型實驗이란 용어로 쓰기로 하며, 費用과 관련된 용어는 holding cost를 維持費, ordering cost를 注文費, set-up cost를 準備費, shortage cost를 不足費라 쓰기로 하고, delivery lag를 注文調達期間, back-log는 未納品注文이라 한다.

## II. 亂數生成과 몬테칼로法

### 1. 問題의 概要

이제 實驗하고자 하는 在庫管理分野의 問題의 概要를 설명하고자 한다.

單一商品을 販賣하고 있는 한 會社를 생각하기로 한다. 다음 n개월에 있어서 매월 얼마나 많은 수준의 在庫를 보유해야 하는가를 決定해야 된다고 가정한다. 매 需要사이의 시간간격은 평균 0.1개월(3일)로서 獨立固有의 確率指數函數(independent, identical dist-

tribution) 를 보인다고 한다. 그리고 需要量  $D$ 는 다음과 같은 確率變數를 (需要사이에는 獨立的)보인다고 한다.

$$D = \left\{ \begin{array}{ll} 1 \text{ 개의 확률 } \frac{1}{6}, & 2 \text{ 개의 확률 } \frac{1}{3} \\ 3 \text{ 개의 확률 } \frac{1}{3}, & 4 \text{ 개의 확률 } \frac{1}{6} \end{array} \right\}$$

매월 첫날 會社는 在庫水準을 점검하여 얼마나 많은 量을 주문할 것인가를 決定한다. 만약 會社가  $Z$ 量 만큼 주문하면 注文費는  $K + iZ$ 만큼 발생한다. 여기에서  $K$ 는 準備費로서 50.0 단위 원이고,  $i$ 는 增分備用인데 1單位당 5.0 원이다. 한 件의 注文이 決定되면 注文調達期間은 0.5 個月에서 1.0 個月( 15 日에서 30 日 )사이를 보이는 均等分布라 한다.

매월 첫날 注文量的 決定은 固定的 ( $s, S$ )在庫模型을 사용하여 決定한다. 즉,  $I$ 를 在庫水準이라 할때  $Z$ 는

$$Z = \begin{cases} S - I & ( I < s \text{인 경우} ) \\ 0 & ( I \geq s \text{인 경우} ) \end{cases}$$

와 같이 된다. 따라서  $I < s$ 이면  $S - I$ 만큼 注文하여  $S$ 만큼 높이고,  $I \geq s$ 이면 注文하지 않는다.

한 件의 需要가 발생하면 즉시 만족시켜 주는데, 만약 需要가 在庫水準보다 높다면 초과된 量 만큼은 未納品注文으로 처리되었다가 후일 注文品이 도착하면 우선 충족된다고 한다. ( 이 경우 새로운 在庫水準은 前水準에서 需要량을 뺀 것인데 負의 整數가 된다 ) 그리고 나머지가 在庫로 창고등에 보관된다.

在庫維持費는 창고비용, 보험료, 세금, 브유비등을 包含하고, 또 機會費用도 包含된다고 한다. 그리고 1單位의 在庫維持費用은 1 個月에 平均 2.5 단위원이 발생한다고 한다. 在庫不足費는 未納品이 있을 때 고객으로 부터 聲價가 떨어지는 것 또는 추가적인 기록 비용등으로서 1單位에 1 個月 平均 8.0 단위원이라고 한다. 이 費用은 임의로 추정한 것인데 일반적으로 在庫維持費보다 높다고 본다.

## 2. 亂需生成

### (1) $U(0, 1)$ 의 亂數生成

위의 問題를 먼저 몬테칼로法으로 실험하기 위해 먼저 平均 0.1의 指數函數亂數, 需要量의 亂數, 注文調達時間인  $U(0.5 \sim 1.0)$ 의 亂數를 생성해야 하는데, 이는  $U(0, 1)$ 의 亂數를 이용함으로써 可能하기 때문에 이를 먼저 生成시키기로 한다.

물론 亂數는 각 컴퓨터마다 고유의 프로그램을 入力시켜 사용하는 것이 일반적이지만, 여기에서는 특별히 함수를 만들어 사용하고자 한다. 여러 방법중에서 혼합항등식방법 (mixed congruential method)로서 먼저 얻은 亂數로 부터 다음 亂數를 계산하여 亂數의 수열을 生成해 간다. 컴퓨터 프로그램은 부록에 표시된 프로그램을 이용하였으며 M이 566927 이므로 亂數의 일련의 길이는 566927 이고 그 이상 진행될 때에는 앞의 數와 똑같이 반복될 것이다. 최초의 亂數를 0.5 로 하고 100 個를 生成하여 제시하였다.

< 표 1 >  $U(0, 1)$ 의 100개의 난수

.00687	.82584	.95692	.40010	.15538	.22276	.60224	.29382
.29751	.61330	.24119	.18118	.43328	.99747	.87377	.06217
.56267	.37257	.49998	.93679	.68570	.74671	.11982	.80504
.80235	.11177	.73329	.66692	.91264	.47047	.34040	.52243
.01656	.82279	.94113	.37157	.11411	.16875	.53549	.21434
.20529	.50834	.12349	.05074	.29010	.84156	.70511	.88078
.36854	.16840	.28037	.70444	.44061	.48888	.84926	.52173
.50631	.80299	.41177	.33266	.56564	.11073	.96792	.13721
.61861	.41210	.51770	.93540	.66520	.70710	.06111	.72721
.70542	.99573	.59815	.51266	.73928	.27800	.12882	.29174
.76676	.55389	.65312	.06445	.78788	.82341	.17105	.83079
.80262	.08657	.68261	.59075	.81100	.34335	.18780	.34435
.81301	.59377	.68662	.09158				

이와같이 얻은 亂數가 진정한 亂數인가는 適合性檢定으로서 판단할 수 있는데 여러가지 檢定중에서  $\chi^2$ -test 를 실시하였다. 檢증을 위한  $\chi^2$  계산은  $\chi^2 = 5.6 < \chi_{0.05}^{2(9)} = 16.9190$  으로서 亂數로서 適合한 것으로 나타났다.

(2) 指數函數(平均3日)亂數의 生成

需要와 需要사이의 간격은 평균 0.1個月(3日)이므로, 먼저  $U \sim U(0, 1)$ 을 生成하고  $LOC3 = LOG(U) * (-MEAN)$ 하면 된다. 참고로 30일을 곱하였고, 이의 프로그래프와 처음 100개의 亂數를 제시하였다. 이 亂數도 區間을 1日로 하여  $\chi^2$ -test 를 실시한 결과  $\chi^2 = 8.65 < \chi_{0.05}^{2(9)} = 19.9190$  (d. f. = 11-1-1 = 9)와 같이 되어 亂數로 適合하였다.

(3) 需要量亂數의 生成

주어진 問題에 의하여 需要量은 1, 2, 3, 4이고 각각의 確率은 0.16667, 0.33333,

< 표 2 > 지수함수(평균 3일)의 100개의 난수

14.94122	0.57405	0.13211	2.74815	5.58573	4.50503	1.52130	3.67432
3.63692	1.46671	4.26652	5.12477	2.50914	0.00760	0.40482	8.33367
1.72518	2.94029	2.07956	0.19589	1.13195	0.87624	6.36523	0.65060
0.66061	6.57383	0.93062	1.21527	0.27423	2.26207	3.23292	1.94780
12.30223	0.58515	0.18202	2.97006	6.51180	5.33799	1.87369	4.62056
4.75002	2.02983	6.27480	8.94295	3.71260	0.51751	1.04818	0.38086
2.99463	5.34417	3.81493	1.05106	2.45879	2.14690	0.49018	1.95180
2.04181	0.65823	2.66184	3.30192	1.70937	6.60194	0.09781	5.96867
1.44086	2.65945	1.97508	0.20035	1.22300	1.03974	8.38536	0.95560
1.04687	0.01283	1.54176	2.00442	0.90624	3.84044	6.14810	3.69569
0.79673	1.77237	1.27800	8.22575	0.71523	0.58290	5.29744	0.55615
0.65960	7.34055	1.14550	1.57907	0.62846	3.20701	5.01711	3.19825
0.62104	1.56381	1.12791	7.17149				

0.33333, 0.16667 이었다. 따라서 앞에서 얻은 亂數  $U(0,1)$ 을 갖고,  $U$ 가 0 ~ 0.16667이면 1,  $U$ 가 0.16668 ~ 0.50000이면 2,  $U$ 가 0.50001 ~ 0.83333이면 3,  $U$ 가 0.83333 ~ 1.00000이면 4가 되도록 하여 亂數를 生成시킨다. 프로그램과 처음 100개의 亂數를 제시하였다. 참고로  $\chi^2$ -test를 실시한 결과  $\chi^2 = 0.566 < \chi^2_{0.05}^{(3)} = 7.815$ 로서 적합하였다.

< 표 3 > 수요량 100 회의 난수

3	3	3	4	3	3	3	2	2	2	1	4	1	2	2	2	3	2
2	3	2	2	3	2	1	2	4	3	4	2	1	1	3	2	2	3
2	2	2	1	4	1	3	2	3	1	3	4	2	4	4	2	4	4
1	3	3	3	2	1	2	3	2	3	4	3	3	4	3	2	3	2
1	2	4	3	3	2	4	1	2	1	1	2	1	4	2	3	3	3
2	1	2	3	2	2	3	3	2	3								

(4) 注文調達期間  $U(0.5 \sim 1.0)$ 의 亂數生成

주어진 問題에 의하여 注文調達期間은 0.5 ~ 1.0 個月(15日~30日)의 均等分布이므로, 역시 처음의 亂數  $U(0,1)$ 를 이용하여  $U(0.5 \sim 1.0)$ 의 亂數를 生成시킬 수 있다. 즉  $UNIFORM = 0.5 + (X * (1.0 - 0.5))$ 로 놓고, 30日을 곱하였다. 이의 프로그램과 처음 亂數 100個를 제시하였다.

참고로  $\chi^2$ -test를 실시한 결과(간격 3日)  $\chi^2 = 0.80 < \chi^2_{0.05}^{(8)} = 9.48773$ 으로 적합하였다.

< 표 4 >  $U(15, 30)$ 의 100개의 난수

15.10307	27.38766	29.35379	21.00143	17.33063	18.34134	24.03357
19.40735	19.46265	24.19946	18.61783	17.71771	21.49911	29.96207
28.10654	15.93254	23.44006	20.62912	22.49969	29.05182	25.28546

26,20062	16,79733	27,07556	27,03531	16,67659	25,99940	25,00375
28,68962	22,05702	20,10596	22,83641	15,24840	27,34190	29,11696
20,57353	16,71162	17,53125	23,03241	18,21510	18,07932	22,62506
16,85234	15,76114	19,35147	27,62332	25,57671	28,21162	20,52808
17,52605	19,20555	25,56657	21,60913	22,33322	27,73885	22,82599
22,59467	27,04486	21,17659	19,98985	23,48465	16,66096	29,51880
17,05818	24,27908	21,18152	22,76547	29,03096	24,97798	25,60652
15,91661	25,90820	25,58134	29,93600	23,97218	22,68991	26,08916
19,16994	16,93224	19,37607	26,50143	23,30832	24,79674	15,96670
26,81816	27,35117	17,56570	27,46176	27,03935	16,29848	25,23921
23,86130	27,16501	20,15024	17,81700	20,16530	27,19511	23,90646
25,29935	16,37375					

### 3. 몬테칼로法

먼저 在庫水準의 增減狀態를 실험하기 위해 册床위 手作業으로 간단한 몬테칼로 實驗을 한다. 亂數는 앞에서 生成한 것을 모두 11 번째 것부터 이용하기로 하고, 在庫模型은 (20, 80) 政策이라 가정한다. 첫날은 7월 1 일이라 하고, 주어진 기초 在庫水準은 60 單位라 한다. 그리고 실험의 편리와 연속성을 위해 공휴일이나 일요일의 사상을 무시하고, 모두 실험속에 包含시킨다.

부연할 것은 時間 간격은 소숫점 2 자리에서 버렸으며, 그림의 作圖가 어려워 需要가 발생하여 在庫가 감소하는 것은 그날의 끝에 감소하는 것으로 간주하였다. 이와같이 가정하여 처음 6 개월간 日別 在庫水準의 變化를 보인 것이 그림과 같으며 9월과 12월에 在庫不足이 발생했으며, 注文은 9월 1일과 12월 1일에 있었다.

또 12월 1 個月의 在庫費用을 계산한 결과 表에서 보는 바와 같이 注文費, 維持費, 不足費를 합하여 538,580 원이다.



12 月の 계산에서 보인 것과 같이 앞의 그림을 이용하여 6 개월간의 月平均 在庫費用 (ACOST)를 구하고자 한다. 먼저 月平均維持費를 다음과 같이 구한다. 1 個月의 기간을  $t = 1.0$ 이라 놓으면 12 月 31 日까지는  $t = 0.0$ 에서  $t = 6.0$  까지 이므로 在庫水準의 총면적은  $\int_{0.0}^{6.0} I^+(t) \cdot dt$  가 될 것이다. 따라서 1 個月의 平均 維持費 AHLDC는,

$$\begin{aligned} \text{AHLDC} &= h \cdot \frac{\int_{0.0}^{6.0} I^+(t) dt}{6} \\ &= 2.5 \frac{(50.1905 + 19.8064 + 27.1333 + 50.6129 + 16.2000 + 21.6774)}{6} \\ &= 77.3627 \text{ (단위원)} \dots\dots\dots (1) \end{aligned}$$

가 된다.

다음, 月平均 在庫不足費用은 在庫不足의 총면적이  $\int_{0.0}^{6.0} I^-(t) \cdot dt$  가 될 것이므로,

1 個月의 平均 不足費 ASHRC는,

$$\begin{aligned} \text{ASHRC} &= \pi \cdot \frac{\int_{0.0}^{6.0} I^-(t) \cdot dt}{6} \\ &= 8.0 \frac{(0.2667 + 5.54838)}{6} \\ &= 7.7534 \text{ (단위원)} \dots\dots\dots (2) \end{aligned}$$

와 같이 된다.

그리고 月平均 注文費는 9 月の 注文費가  $50.0 + 5.0(69 \text{ 단위}) = 395.0$  원 이고, 12 月の 注文費가 440.0 원이므로 AORDC는,

$$\begin{aligned} \text{AORDC} &= (395.0 + 440.0) / 6 \\ &= 139.1667 \text{ (단위 원)} \dots\dots\dots (3) \end{aligned}$$

이 된다. 따라서 月平均 在庫費用 ACOST는

$$\begin{aligned}
 \text{ACOST} &= \text{AHLDC} + \text{AORDC} + \text{ASHRC} \\
 &= 77.3627 + 139.1667 + 7.7534 \\
 &= 224.2828 \text{ (단위 원)} \dots\dots\dots (4)
 \end{aligned}$$

가 된다.

### Ⅲ. 포오프트란 프로그램

#### 1. 事象과 變數와 副프로그램

프로그램을 설명하기 전에 먼저 언급할 점들은 다음과 같다. 이 在庫管理의 模型實驗에서 狀態變數( state variables)는 在庫水準  $I(t)$  와, 會社가 供給者에 주문해 놓은 注文量, 가장 最近에 在庫水準을 變化시킨 事象이 발생한 時間( 이는 함수  $I^+(t)$  와  $I^-(t)$  )의 아랫부분 면적을 계산하는 데 必要 함 )이다. 그리고 이 모델에서는 다음 4가지 類型的 事象( events )을 사용하고자 한다.

내 용	事象의 類型
주문한 것이 공급자로 부터 회사에 도착하는 것	1
수요가 고객으로 부터 발생하는 것	2
모형실험이 n개월후에 끝나는 것	3
매월초에 재고를 조사하는 것( 주문할 경우가 있음 )	4

模型實驗의 끝으로 事象 4보다 事象 3을 擇했는데, 이는  $\text{TIME} = 120$ 에서 「模型實驗의 끝」과 「在庫調査」事象이 동시에 일어날 것이며 이때 「實驗의 끝」事象이 먼저 실행 되도록 하기 위한 것이다. 둘 또는 그 이상의 事象이 동시에 발생하였 ( 이를 타임타이 time tie라 한다 )을 때에는 TIMING이 적은 값의 數를 먼저 擇하기 때문이다.

모델의 컴퓨터 프로그램은 主프로그램, 副프로그램 그리고 포오프트란 變數들로 구성되어 있는데 먼저 變數들을 나열하면 다음과 같다.

모수 및 변수	정의
<u>투 입 모 수 ( input parameters )</u>	
BIGS	( s, S ) 재고모형에서의 뒤의 상한치
H	h, 재고유지비에서의 승수
INCRMC	i, 주문시 개당 증분비용
MDEMDT	수요간격의 평균시간( 지수함수에서의 평균 )
NMNTHS	모형실험의 기간( 개월 수 )
NPOLCY	재고정책의 가지수
NVALUE	수요량의 가지수
PI	재고부족비용의 승수
PROBD( I )	수요 demand $\leq I$ 일 확률
SETUPC	K, 한건의 주문을 위한 준비비
SMALLS	( s, S ) 재고모형에서의 앞의 하한치
<u>모델화 변수 ( modeling variables )</u>	
AMINUS	함수 $I^-(t)$ 아래부분의 면적
AMOUNT	Z, 회사가 공급자한테 주문한 주문량
APLUS	함수 $I^+(t)$ 의 아래부분의 면적
DSIZE	특정 수요량
INVLEV	$I(t)$ , 재고수준
NEVNTS	사상유형의 가지수( 여기에서는 4 가지 )
NEXT	다음 사상이 일어날 사상유형( 1, 2, 3 또는 4 )
TIME	모형실험의 시계
TLEVNT	최근에 재고수준을 변화시킨 사상의 시간
TNE( I )	사상유형 I ( I= 1, 2, 3, 4 )가 발생한 시간
TORDC	총 주문비용
TSLE	최근의 사상이 발생한 이후부터의 시간

산출변수( output variables )

ACOST	월평균 총비용
AHLDC	월평균 유지비
AORDC	월평균 주문비
ASHRC	월평균 부족비

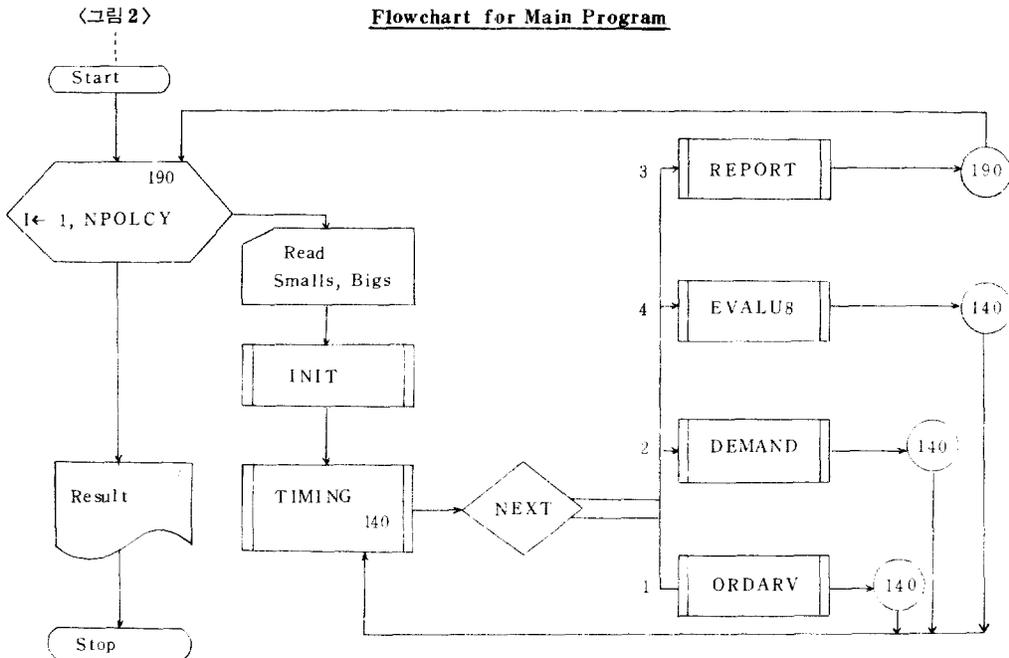
부프로그램은 앞의 亂數生成에서 보인 함수등을 포함하여 11개로 되어 있는데, 다음과 같은 목적을 갖고 있다.

부프로그램	목적
INIT	루우틴을 시작한다.
TIMING	타이밍 루우틴
ORDARV	사상 1을 진행시키는 사상루우틴
DEMAND	사상 2를 진행시키는 사상루우틴
REPORT	사상 3을 진행시키는 사상루우틴
EVALU8	사상 4를 진행시키는 사상루우틴
UPDATE	함수 $I^+(t)$ 와 $I^-(t)$ 아래 면적을 계산하는 부프로그램 인데 재고에 변화가 오거나 실험이 끝날때 한다.
EXPON(RMEAN)	평균 RMEAN을 가진 확률지수함수의 변수를 생성시키는 함수
RANDI(Z)	1과 NVALUE 사이의 정수 난수를 생성시키는 함수. 분포함수 $PROBD(I)$ ( $I = 1, 2, \dots, NVALUE$ )에 따르고 Z는 의제변수이다. NVALUE의 값과 $PROBD(I)$ 는 주프로그램에 넣어지고 주프로그램에 다음과 같은 공통블럭을 설정함으로 RANDI에 전달된다. $COMMON/RANDOM/NVALUE, PROBD(25)$
UNIFORM(A, B)	구간 $[A, B]$ 를 갖는 연속 균등분포의 확률변수를 생성시키는 함수. 실수이며 $A < B$ .
RANUN(X)	구간 $[0, 1]$ 을 갖는 연속 균등분포의 확률변수를 생성시키는 함수. (3.2.1과 같음)

## 2. 主프로그램

主프로그램은 공통블럭 MODEL과 RANDOM으로 시작하고, NEVNTS를 4로 놓고, 투입모수를 읽어 들인다. 고려하고 있는 (s, S) 在庫模型의 각 정책마다 副프로그램 REPORT를 호출해야 하므로 컴퓨터 레포트의 題目을 REPORT보다 主프로그램에서 프린트한다.

다음 각각의 在庫模型의 政策마다 데이터 카드로 부터 s와 S 값을 읽어 들이고, 副프로그램 INIT를 호출함으로써 실험이 시작된다. 이의 順序圖가 그림에 제시되어 있다.



그다음 발생할 事象의 類型 즉 NEXT는 副프로그램 TIMING을 부름으로써 결정된다. NEXT는 다음의 알맞는 事象을 통제하는데 사용된다. 즉 計算型 GO TO 명령문으로써 만약 NEXT=3 이라면 REPORT가 호출되고, 지금까지 시행되던 바로 그 政策의 실험은 끝나게 된다.

이와같이 하여 재고정책의 가지수 만큼 실험이 진행된다.

### 3. 副프로그램

副프로그램 INIT는 실험을 시작하는 프로그램인데 模型實驗의 時計를 TIME= 0. 로 놓고 狀態變數들을 시작화하고, TORDC, APLUS, AMINUS를 모두 0. 으로 놓고 시작한다. 첫번째 在庫調査를 TIME=0. 에서 진행되도록 하였는데 이는 초기 재고수준이 s보다 적을 수 있기 때문이다.

副프로그램 TIMING에서는 TNE (1), TNE (2), ..., TNE(NEVNTS) 를 비교하여 그중 시간이 가장 적은 事象을 골라 NEXT가 되게 한다. 타이의 경우에는 수값이 적은 쪽의 事象을 선택한다. 그리고 나면 模型實驗의 時計는 선택된 事象의 시간만큼 즉 TNE (NEXT) 까지 흘러 간다. 만약 事象리스트가 비어있다면 즉 NEXT=0 라면 에러 메시지 「 event list empty」가 인쇄되고 실험은 끝나게 된다.

副프로그램 ORDARV에서는 먼저 副프로그램 UPDATE를 불러서 함수  $I^+(t)$   $I^-(t)$  의 면적을 最新化(update) 한다. 왜냐하면 바로 이 事象이 발생하는 시기에 在庫水準이 변화할 것이기 때문이다. 그러면 在庫水準 INVLEV는 注文했던 量 AMOUNT 만큼 증가할 것이고 다음 注文이 도착할 시간인 TNE(1)은 다른 注文이 밀려있지 않기 때문에  $10^{30}$ 으로 놓는다.

副프로그램 DEMAND는 需要가 발생했을 때 호출되는데, 이때 在庫水準이 變化한 것이기 때문에 먼저 副프로그램 UPDATE를 불러 函數  $I^+(t)$  와  $I^-(t)$  를 最新化한다. 需要量 DSIZ는 函數 RANDI에서 亂數를 생성시킴으로써 결정되며, 다음에 在庫水準에서 이를 減한다. 이때 在庫水準이 負가 될 수도 있는데 이는 未納品으로 처리된다. 다음 需要가 발생할 시간인 TNE(2)는 현재의 實驗時計에 평균 MDEMDT를 가진 指數 函數 亂數를 生成시켜 더해 놓으면 된다.

副프로그램 EVALU 8은 매월초 在庫調査를 할 때에 (필요한 경우 注文함) 호출되는데, 만약 在庫水準 INVLEV가 政策의 下限值인 s보다 작다면 注文하는데, 量은 AM-

OUNT=BIGS-INVLEV가 된다. 그리고 費用은  $SETUPC+(INCRMC \times AMO-UNT)$ 가 된다. 이때 注文이 도착한 時間인 TNE(1)은 模型實驗의 현재 시간에 均等分布  $U(0.5 \sim 1.0)$ 의 亂數를 生成시켜 더해 놓으면 된다. 다음 注文을 하든 안하든 在庫調查를 할 다음시간인 TNE(4)는  $TIME+1.0$ 으로 놓으면 된다.

副프로그램 REPORT에서는 먼저 副프로그램 UPDATE를 불러서 在庫水準을 變化시킨 마지막 事象으로부터 實驗이 끝난 시간인  $TIME=NMNTHS$ 까지의 函數  $I^+(t)$ 와  $I^-(t)$ 의 면적을 最新化한다. 그리고 지금까지의 결과를 계산하여 ACOST, AO-RDC, AHLDC, ASHRC를 인쇄하도록 한다.

副프로그램 UPDATE는 事象의 한 類型은 아니지만 函數  $I^+(t)$ 와  $I^-(t)$ 의 면적을 最新化하기 위해, ORDARV와 DEMAND 및 REPORT로부터 호출되는 데 똑같은 블럭코드를 세번이나 쓰는 것을 대신해준다. 만약 앞의 시간 간격에서 在庫水準이 負이었다면 函數  $I^-(t)$ 가 最新化되고,  $I^+(t)$ 는 變化하지 않는다. 만약 在庫水準이 零이라면 둘다 變化하지 않고, 陽이라면  $I^+(t)$ 가 最新化되고  $I^-(t)$ 는 變化하지 않는다.

다음 函數들은 앞의 몬테칼로法에서 이미 설명하였으므로 부연하지 않는다. EXPON(RMEAN)은 수요간격의 시간을, RANDI(Z)는 需要量을, UNIFRM(A,B)는 注文調達期間의 亂數를, RANUN(X)는  $U(0,1)$ 의 亂數를 生成한다.

#### IV. 結 論 : 實驗結果와 分析

초기의 주어진 在庫水準을 60單位라 하고, 實驗의 길이를 120個月(10年,  $t=120$ )로 한 結果를 (s,S) 政策別, 月平均費用, 注文費, 維持費, 不足費로 區分하여 인쇄하였다.

表에서 보면 (20,40) 政策이 가장 費用이 적은 데, 그 구성내용은 維持費가 가장 적은 대신에 不足費가 가장 크다.

이 推定費用들은 큰 分散을 가질 수 있기 때문에 우리가 원하는 진정한 推定値와는 어느 정도 차이가 있을 수 있다. 그리고 다른  $U(0,1)$ 의 確率亂數를 사용하여 實驗을 여러번 시행하면 여기에 제시된 費用과 다를 것이다. 模型實驗의 길이가 고정되어 있다면 實驗을 한번만 한다는 것은 적절치 못할 것이다. 따라서 實驗을 몇번해야 되는가, 서

< 표 5 >

컴퓨터 레포트

SINGLE PRODUCT INVENTORY SYSTEM

INITIAL INVENTORY LEVEL 60 ITEMS  
 NUMBER OF DEMAND SIZE 4  
 DISTRIBUTION FUNCTION OF DEMAND SIZES 0.167 0.500  
 0.833 1.000  
 MEAN INTERDEMAND TIME 0.10 MONTHS  
 LENGTH OF THE SIMULATION 120 MONTHS  
 K=50.0 I=5.0 H=2.5 PI=8.0

POLICY	AVERAGE COST	AVG ORDERING COST	AVG HOLDING COST	AVG SHORT-AGE COST
( 20, 40 )	207.84	158.25	24.77	24.82
( 20, 60 )	215.01	148.12	45.39	21.49
( 20, 80 )	221.16	140.67	66.33	14.16
( 20, 100 )	239.57	135.17	95.54	8.86
( 40, 60 )	226.64	158.71	64.70	3.24
( 40, 80 )	236.85	146.37	88.24	2.23
( 40, 100 )	255.22	140.83	113.95	0.44
( 60, 80 )	275.42	158.37	117.03	0.01
( 60, 100 )	284.05	148.71	135.02	0.32

로 다른 數값이 나왔을 때 어떻게 해석해야 할 것인가를論해야 할 것 같다. 이 분야는 참고문헌 Gordon, Law, Shannon 등에 부분적으로 소개되어 있는데, 필자가 다음 기회의 연구과제로 남겨 놓는다.

또 投入母數를 달리해서 實驗했을 경우 어떠한 結果가 나오는가를 보아야 하고, 이로써 模型實驗의 長·短點을 그리고 그 限界를 알아야 할 것이다.

## 〈컴퓨터 프로그램 리스트〉

```

          FACOM BOS/V5  FORTRAN S  -770127- V01-L14
OPTIONS IN EFFECT (FORTS1)
OBJECT,NOSTACK,SOURCE,NOMAP,NOISN,EBCDIC,NOALTODBL,NOSEQUENCE,NOASTERISK,NODENSEPRINT,
OPT(0),FMTAREA( 256),FLAG(W),DEBUG(STD)

C
C
C *** MAIN PROGRAM.
C
0001      INTEGER AMOUNT, BIGS, INITIL, INVLEV, NEVNTS, NEXT, NMNTHS, SMALLS
0002      INTEGER I, NPOLCY
0003      REAL AMINUS, APLUS, H, INCRMC, MDEMMDT, PI, SETUPC, TIME, TLEVNT,
+TNE(4), TORDC
0004      COMMON /MODEL/ AMINUS, AMOUNT, APLUS, BIGS, H, INCRMC, INITIL,
+INVLEV, MDEMMDT, NEVNTS, NEXT, NMNTHS, PI, SETUPC, SMALLS,
+TIME, TLEVNT, TNE, TORDC
0005      COMMON /RANDOM/ NVALUE, PROBD(25)

C
C *** SPECIFY THE NUMBER OF EVENT FOR THE TIMING ROUTE.
0006      NEVNTS=4

C
C *** READ INPUT PARAMETERS.
0007      READ(5,10) INITIL,NMNTHS,NPOLCY,NVALUE
0008      10  FORMAT(4I10)
0009      READ(5,20)MDEMMDT,SETUPC,INCRMC,H,PI
0010      20  FORMAT(5F10.2)
0011      READ(5,30)(PROBD(I),I=1,NVALUE)
0012      30  FORMAT(8F10.5)

C
C *** PRINT REPORT HEADING
0013      WRITE(6,40)
0014      40  FORMAT(1H1, 5X, 'SINGLE PRODUCT INVENTORY SYSTEM')
0015      WRITE(6,50) INITIL
0016      50  FORMAT(1H1, 5X, 'INITIAL INVENTORY LEVEL', 21X, I3, ' ITEMS')
0017      WRITE(6,60) NVALUE
0018      60  FORMAT(1H0, 5X, 'NUMBER OF DEMAND SIZE', 22X, I3)
0019      WRITE(6,70) (PROBD(I),I=1,NVALUE)
0020      70  FORMAT(1H0, 5X, ' DISTRIBUTION FUNCTION OF DEMAND SIZES', 5X,
+8( F5.3, 3X))
0021      WRITE(6,80)MDEMMDT
0022      80  FORMAT(1H0, 5X, 'MEAN INTERDEMAND TIME', 21X, F5.2, ' MONTHS')
0023      WRITE(6,90) NMNTHS
0024      90  FORMAT(1H0, 5X, 'LENGTH OF THE SIMULATION', 20X, I3, ' MONTHS')
0025      WRITE(6,100)SETUPC,INCRMC,H,PI
0026      100 FORMAT(1H0, 5X, 'K = ', F5.1, 3X, 'I = ', F5.1, 3X, 'H = ',
+F5.1, 3X, 'F' = ', F5.1)
0027      WRITE(6,110)
0028      110 FORMAT(//)
0029      WRITE(6,120)
0030      120 FORMAT(1H0, 7X, 'POLICY', 6X, 'AVERAGE COST', 5X,
+'AVERAGE ORDERING COST', 5X, 'AVERAGE HOLDING COST', 5X,
+'AVERAGE SHORTAGE COST')

C
C *** RUN THE SIMULATION VARYING THE INVENTRRY POLICY.
0031      DO 190 I=1, NPOLCY

C
C *** READ THE INVENTORY POLICY.
0032      READ(5,130) SMALLS,BIGS
0033      130 FORMAT(2I10)

C
C *** INITIALIZE THE SIMULATION.
0034      CALL INJT

C
C *** DETERMINE THE NEXT EVENT.
0035      140 CALL TIMING

C
C *** CALL APPROPRIATE EVENT ROUTINE.

```

```

0036          GO TO (150, 160, 180, 170), NEXT
0037          150 CALL ORDARV
0038          GO TO 140
0039          160 CALL DEMAND
0040          GO TO 140
0041          170 CALL EVALU8
0042          GO TO 140
0043          180 CALL REPORT
0044          190 CONTINUE
0045          WRITE(6, 200)
0046          200 FORMAT(1H1)
0047          STOP
0048          END

0001          SUBROUTINE INIT
0002          INTEGER AMOUNT, BIGS, INITIL, INVLEV, NEVNTS, NEXT, NMNTHS, SMALLS
0003          REAL AMINUS, APLUS, H, INCRMC, MDEMDT, PI, SETUPC, TIME, TLEVNT,
          +TNE(4), TORDC
0004          COMMON /MODEL/ AMINUS, AMOUNT, APLUS, BIGS, H, INCRMC, INITIL,
          +INVLEV, MDEMDT, NEVNTS, NEXT, NMNTHS, PI, SETUPC, SMALLS,
          +TIME, TLEVNT, TNE, TORDC
          C
          C *** INITIALIZE THE SIMULATION CLOCK.
0005          TIME=0.
          C
          C *** INITIALIZE THE STATE VARIABLES.
0006          INVLEV=INITIL
0007          TLEVNT=0.
          C
          C *** INITIALIZE THE STATISTICAL COUNTERS.
0008          TORDC=0.
0009          APLUS=0.
0010          AMINUS=0.
          C
          C *** INITIALIZE THE EVENT LIST. SINCE NO ORDER IS OUTSTANDING,
          C *** THE TIME OF THE NEXT ORDER ARRIVAL IS TO 'INFINITY.'
0011          TNE(1)=1.E+30
0012          TNE(2)=EXPON(MDEMDT)
0013          TNE(3)=NMNTHS
0014          TNE(4)=0.
0015          RETURN
0016          END

0001          SUBROUTINE TIMING
0002          INTEGER AMOUNT, BIGS, INITIL, INVLEV, NEVNTS, NEXT, NMNTHS, SMALLS
0003          REAL AMINUS, APLUS, H, INCRMC, MDEMDT, PI, SETUPC, TIME, TLEVNT,
          +TNE(4), TORDC
0004          REAL RMIN
0005          COMMON /MODEL/ AMINUS, AMOUNT, APLUS, BIGS, H, INCRMC, INITIL,
          +INVLEV, MDEMDT, NEVNTS, NEXT, NMNTHS, PI, SETUPC, SMALLS,
          +TIME, TLEVNT, TNE, TORDC
0006          RMIN=1.E+29
0007          NEXT=0
          C
          C *** DETERMINE THE EVENT TYPE OF NEXT EVENT TO OCCUR.
0008          DO 10 I=1, NEVNTS
0009             IF(TNE(I) .GE. RMIN) GO TO 10
0010             RMIN=TNE(I)
0011             NEXT=I
0012          10 CONTINUE
          C
          C *** IF THE EVENT LIST IS EMPTY (I.E., NEXT=0), STOP THE SIMULATION
          C *** OTHERWISE, ADVANCE THE SIMULATION CLOCK.
0013          IF(NEXT .GT. 0) GO TO 30
0014          20 FORMAT(1H1, 5X, 'EVENT LIST EMPTY')
0015          STOP
0016          30 TIME=TNE(NEXT)
0017          RETURN
0018          END

```

```

0001      SUBROUTINE ORDARY
0002      INTEGER AMOUNT, BIGS, INITIL, INVLEV, NEVNTS, NEXT, NMNTHS, SMALLS
0003      REAL AMINUS, APLUS, H, INCRMC, MDEMOT, PI, SETUPC, TIME, TLEVNT,
+TNE(4), TORDC
0004      COMMON /MODEL/ AMINUS, AMOUNT, APLUS, BIGS, H, INCRMC, INITIL,
+INVLEV, MDEMOT, NEVNTS, NEXT, NMNTHS, PI, SETUPC, SMALLS,
+TIME, TLEVNT, TNE, TORDC
      C
      C *** UPDATE 'APLUS' AND 'AMINUS.'
0005      CALL UPDATE
      C
      C *** INCREMENT THE INVENTORY LEVEL BY THE AMOUNT ORDERED
0006      INVLEV=INVLEV+AMOUNT
      C
      C *** SINCE NO ORDER IS NOW OUTSTANDING, SET THE TIME OF THE NEXT
      C *** ORDER ARRIVAL TO 'INFINITY.'
0007      TNE(1)=1.E+30
0008      RETURN
0009      END

0001      SUBROUTINE DEMAND
0002      INTEGER AMOUNT, BIGS, INITIL, INVLEV, NEVNTS, NEXT, NMNTHS, SMALLS
0003      INTEGER DSIZ:
0004      REAL AMINUS, APLUS, H, INCRMC, MDEMOT, PI, SETUPC, TIME, TLEVNT,
+TNE(4), TORDC
0005      COMMON /MODEL/ AMINUS, AMOUNT, APLUS, BIGS, H, INCRMC, INITIL,
+INVLEV, MDEMOT, NEVNTS, NEXT, NMNTHS, PI, SETUPC, SMALLS,
+TIME, TLEVNT, TNE, TORDC
      C
      C *** UPDATE 'APLUS' AND 'AMINUS.'
0006      CALL UPDATE
      C
      C *** GENERATE THE DEMAND SIZE
0007      DSIZ=RANDI(?)
      C
      C *** DECREMENT THE INVENTORY LEVEL BY THE DEMAND SIZE.
0008      INVLEV=INVLEV-DSIZ
      C
      C *** SCHEDULE THE NEXT DEMAND.
0009      TNE(2)=TIME+EXPON(MDEMOT)
0010      RETURN
0011      END

0001      SUBROUTINE EVALUB
0002      INTEGER AMOUNT, BIGS, INITIL, INVLEV, NEVNTS, NEXT, NMNTHS, SMALLS
0003      REAL AMINUS, APLUS, H, INCRMC, MDEMOT, PI, SETUPC, TIME, TLEVNT,
+TNE(4), TORDC
0004      COMMON /MODEL/ AMINUS, AMOUNT, APLUS, BIGS, H, INCRMC, INITIL,
+INVLEV, MDEMOT, NEVNTS, NEXT, NMNTHS, PI, SETUPC, SMALLS,
+TIME, TLEVNT, TNE, TORDC
      C
      C *** IF THE INVENTORY LEVEL IS LESS THEN 'SMALLS,' PLACE AN ORDER
      C *** FOR 'BIGS'-'INVLEV' ITEMS.
0005      IF(INVLEV .GE. SMALLS) GO TO 10
0006      AMOUNT=BIGS-INVLEV
0007      TORDC=TORDC+SETUPC+(INCRMC*AMOUNT)
      C
      C *** SCHEDULE THE ARRIVAL OF THE ORDER.
0008      TNE(1)=TIME+UNIFRM(.5, 1.)
      C
      C *** SCHEDULE THE NEXT INVENTORY EVALUATION.
0009      10 TNE(4)=TIME+1.
0010      RETURN
0011      END

0001      SUBROUTINE REPORT
0002      INTEGER AMOUNT, BIGS, INITIL, INVLEV, NEVNTS, NEXT, NMNTHS, SMALLS
0003      REAL AMINUS, APLUS, H, INCRMC, MDEMOT, PI, SETUPC, TIME, TLEVNT,
+TNE(4), TORDC

```

```

0004 REAL ACOST, AHLDC, AORDC, ASHRC
0005 COMMON /MODEL/ AMINUS, AMOUNT, APLUS, BIGS, H, INCRMC, INITIL,
+INVLEV, MDEMDT, NEVNTS, NEXT, NMNTHS, PI, SETUPC, SMALLS,
+TIME, TLEVNT, TNE, TORDC
C
C *** UPDATE 'APLUS' AND 'AMINUS.'
0006 CALL UPDATE
C
C *** COMPUTE ESTIMATES OF THE DESIRED MEASURES OF PERFORMANCE.
0007 AORDC=TORDC/NMNTHS
0008 AHLDC=H*(APLUS/NMNTHS)
0009 ASHRC=PI*(AMINUS/NMNTHS)
0010 ACOST=AORDC+AHLDC+ASHRC
0011 10 FORMAT(1H0,5X, '( ', I3, ', ', I3, ')', 8X, F6.2, 16X, F6.2, 19X,
+ F6.2, 19X, F6.2)
0012 RETURN
0013 END

0001 SUBROUTINE UPDATE
0002 INTEGER AMOUNT, BIGS, INITIL, INVLEV, NEVNTS, NEXT, NMNTHS, SMALLS
0003 REAL AMINUS, APLUS, H, INCRMC, MDEMDT, PI, SETUPC, TIME, TLEVNT,
+TNE(4), TORDC
0004 REAL TSLE
0005 COMMON /MODEL/ AMINUS, AMOUNT, APLUS, BIGS, H, INCRMC, INITIL,
+INVLEV, MDEMDT, NEVNTS, NEXT, NMNTHS, PI, SETUPC, SMALLS,
+TIME, TLEVNT, TNE, TORDC
C
C *** COMPUTE THE TIME SINCE THE LAST EVENT WHICH CHANGED THE INVENTORY
LEVEL
0006 C ***
0007 TSLE=TIME-TLEVNT
TLEVNT=TIME
C
C *** DETERMINE WHETHER THE INVENTORY LEVEL DURING THE PREVIOUS
INTERVAL WAS NEGATIVE, ZERO, OR POSITIVE.
0008 C ***
IF(INVLEV)10, 20, 30
C
C *** SINCE THE INVENTORY LEVEL DURING THE PREVIOUS INTERVAL WAS
NEGATIVE, UPDATE 'AMINUS'
0009 10 AMINUS=AMINUS+(-INVLEV*TSLE)
C
C *** THE INVENTORY LEVEL DURING THE PREVIOUS INTERVAL WAS ZERO.
0010 20 RETURN
C
C *** SINCE THE INVENTORY LEVEL DURING THE PREVIOUS INTERVAL WAS
POSITIVE, UPDATE 'APLUS.'
0011 30 APLUS=APLUS+(INVLEV*TSLE)
0012 RETURN
0013 END

0001 FUNCTION EXPON(RMEAN)
0002 REAL RMEAN, U
0003 DATA Z / 0.5 /
C
C *** GENERATE A U(0, 1) RANDOM VARIABLE.
0004 U=RANUN(Z)
C
C *** GENERATE AN EXPONENTIAL RANDOM VARIABLE WITH MEAN RMEAN
EXPON=LOG(U)*(-RMEAN)
0005 RETURN
0006 END
0007

0001 FUNCTION RAN(I(Z))
0002 INTEGER I, N1
0003 REAL U
0004 COMMON /RANDOM/ NVALUE, PROBD(25)
C
C *** GENERATE A U(0, 1) RANDOM VARIABLE.
0005 U=RANUN(Z)
C
C *** GENERATE A RANDOM INTEGER BETWEEN 1 AND NVALUE IN ACCORDANCE
WITH DISTRIBUTION FUNCTION 'PROBD.'
0006 N1=NVALUE-1

```

```

0007          DO 10 I=1, N1
0008             IF(U .GE. PROBD(I)) GO TO 10
0009             RANDI=I
0010             RETURN
0011          10 CONTINUE
0012             RANDI=NVALUE
0013             RETURN
0014             END

0001          FUNCTION UNIFRM(A, B)
0002             REAL A, B, U
          C
          C *** GENERATE A U(0, 1) RANDOM VARIABLE.
0003             U=RANUN(Z)
          C
          C *** GENERATE A U(A,B) RANDOM VARIABLE.
0004             UNIFRM=A+(U*(B-A))
0005             RETURN
0006             END

0001          FUNCTION RANUN(X)
0002             REAL X, RANUN
          C
          C *** GENERATE RANDOM NUMBER
0003             DATA K, J, M, RM / 5701, 3612, 566927, 566927.0 /
0004             IX=INT(X*RM)
0005             IRAND=MOD(J+IX*K, M)
0006             RAND=IRAND
0007             X=(RAND+0.5)/RM
0008             RANUN=X
0009             RETURN
0010             END

```

참 고 문 헌

- Bohl, M., *Information Processing* 3rd ed., Science Research Associates Inc., Chicago, 1980.
- Chirlian, P. M., *Microsoft Fortran*, Dilithium Press, Oregon, 1981.
- Fishman, G. S., *Principles of Discrete Event Simulation*, Wiley, New York, 1978.
- Forrester, J. W., *Industrial Dynamics*, The M. I. T. Press, Massachusetts, 1961, pp.73-92.
- Gillet, B. E., *Introduction to Operations Research; A Computer-Oriented Algrithm Approach*, Mcgraw-Hill Book Co., 1976, pp.497-563.
- Gordon, G., *System Simulation* 2nd ed., Prentice-Hall, New Jersey 1978.
- Hillier, F. S. and Lieberman, G. J., *Introduction to Operations Research* 2nd ed., Holden-Day, San Francisco, 1974, pp. 473-533, pp.620-56.
- Knuth, D. E., *The Art of Computer Programming*, Vol.1&2, Addison-Wesley, Mass., 1975.
- Law, A. M. and Kelton, W. D., *Simulation Modeling and Analysis*, McGraw-Hill Book Co., 1983.
- Levin, R.I. and Kirkpatrick, *Quantitative Approaches to Management*, McGraw-Hill Kogakusha, Ltd., Tokyo, 1978, pp.428-52.

- Meissner, L. P., *Fortran 77; Featuring Structured Programming*, Addison-Wesley, Mass., 1980.
- Mitchell and Gauthier, Assoc., Inc., *Advanced Continuous Simulation Language user guide/reference manual*, 3rd ed., Mass., 1980.
- Ross, S. M., *A First Course in Probability*, Macmillan Pub. Co. Inc., New York, 1976.
- , *Introduction to Probability Models*, Academic Press, New York, 1980, pp.161-96.
- Ryan, T. A. and Joiner, B. L. and Ryan, B. F., *MINITAB; Student Handbook*, Duxbury Press, Mass., 1976.
- Shannon, R. E., *Systems Simulation; The Art and Science*, Prentice-Hall, New Jersey, 1975.
- Schriber, T. J., *Simulation Using GPSS*, John Wiley & Sons, New York, 1974, pp.277-83.
- Taha, H. A., *Operations Research; An Introduction* 2nd ed., Macmillan Pub. Co. Inc., New York, 1976, pp.438-500.
- Thierauf, R. L. and Klekamp, R. C., *Decision Making Through Operations Research*, reprinted by Tower Press at Seoul, 1976, pp.449-91.
- Wagner H. M., *Principles of Management Science*, Prentice-Hall, New Jersey, 1970, pp.443-574.

## 附 錄

## 韓國經營學會

編輯方針 .....	235
編輯委員會 規則 .....	237



## 「經營學研究」의 編輯方針

### 「經營學研究」編輯委員會

모든 學術論文誌가 기본적으로 요구하는 것은 掲載되는 각 論文이 그 分野의 學術的 知識의 축적에 貢獻해야 한다는 점이다. 學術的 貢獻이란 (1) 그 分野의 새로운 問題를 인식하고 있을 수 있는 解答을 제시하는 일, (2) 새로운 研究技法을 創案하여 既存問題를 해결하는데 적용하는 일, (3) 한 分野의 發展過程에 대한 史的考察, 그리고 (4) 과거에 表明되었던 主張들을 比較分析하여 새로운 討議의 가능성을 발견하는 일 등을 말한다. 한편 知識은 立證되지 않은 主張에 의해서는 產出되지 않기 때문에 憶說로 부터 論證을, 虛勢로 부터 合理的인 論說을, 些少한 것으로 부터 本質을, 그리고 不完全한 技法으로 부터 擁護할 수 있는 研究方法들을 區分하는 일이 學術誌가 해야 할 役割이다. 그 밖에 다른 學術誌와는 달리 學會誌는 會員들에게 幅 넓은 學問討論에의 參與機會는 물론 그네들의 教育活動에 도움이 될 學習資料를 효과적으로 제공할 임무를 지닌다.

위와 같은 諸事實을 염두에 두고 本 編輯委員會는 앞으로 學會誌를 다음과 같이 편집해 나갈 方針이다.

(1) 分科學別로 선임된 編輯委員들에 의한 엄격한 綜合審査가 이루어 지고, 필요에 따라 內容의 修正이 論者에게 指示될 것이다. 審査基準은 (1) 研究主題의 重要度 (2) 研究方法의 妥當性 (3) 內容의 創意性, 檢證性 및 應用性 (4) 論文의 意思傳達의 效果, 그리고 (5) 기타 編輯技術上의 要件에의 符合性등이다.

(2) 紙面이 研究論文, 研究「노트」 및 通信欄으로 구분된다. 經營學의 각 分科學의 知識에 學問的으로 貢獻할 수 있는 論文을 研究論文欄에 掲載하며, 會員들의 教育活動에 도움이 될 學習資料로서의 價値가 認定되는 研究物을 研究「노트」欄에 掲載하며, 그리고 通信欄은 會員들에게 幅 넓은 學問討論에의 參與機會를 제공하는 자리로 이용한다.

(3) 編輯委員會는 執筆 중에 있거나 執筆하고자 하는 論文의 主題를 미리 會員들로 부터 通報 받아 이를 검토하여 執筆을 승인하고 그 사실을 學會誌나 學會報에 발표한다. 이러한 일은 會員들의 同·主題에 대한 중복된 研究를 피하는데 그리고 會員들로 하여금 論文執筆意欲을 갖게 하는 데 效果的인 일 것이다.

(4) 掲載의 優先權은 처음으로 寄稿한 會員의 論文과 그리고 共同으로 執筆된 論文에 주어진다.

(5) 掲載되는 論文은 國文記述을 원칙으로 하여 論文의 意思傳達效果를 높인다.

## 「경영학연구」편집위원회 규칙

### 목 적

제 1 조(목적) : 본 규칙은 「경영학 연구」편집위원회(이하 위원회라 칭한다)의 활동 기준과 절차를 규정함을 목적으로 한다.

### 위 원 회

제 2 조(임무) : 위원회는 한국경영학회 회원 및 기타 기고자로 하여금 우수한 연구논문을 「경영학연구」에 기고하도록 권장하고, 동지에 게재될 논문을 선정하며, 동지의 편집과 관련된 여타의 임무를 수행한다.

제 3 조(조직) : 위원회는 10명 이내로 구성하되 위원장, 편집간사, 재정간사 각 1인을 둔다. 재정간사는 한국경영학회의 상임이사가 겸임한다.

제 4 조(위원의 위촉 및 임기) : 위원은 경영학회 회장이 위촉하며, 각위원의 임기는 2년으로 한다. (단, 재정간사는 1년)

제 5 조(회의) : 위원회는 위원장이 소집하며, 별도 규정하지 않는 한 재적위원 과반수의 찬성으로 의결한다.

### 편 집 절 차

제 6 조(원고 모집에 관한 공지) : 위원장은 「경영학연구」의 원고를 모집하는 사실에 관해 경영학회 회원 및 기타 기고자에게 널리 공지하여야 한다.

제 7 조(심사위촉) : 위원장은 기고된 연구논문을 그 서자가 식별되지 않도록 조치한 후 전공별 해당위원에게 심사기한을 명시 배부하여 심사를 위촉한다.

제 8 조(위원심사) : 논문의 심사를 위촉받은 위원은 정해진 기간내에 심사를 완료하고 심사 의견서를 작성 그 결과를 위원장에게 회송하여야 한다.

제 9 조(논문의 수정) : 다소의 수정 또는 보완으로 채택이 가능하다고 판단되는 논문에 대하여 해당 위원은 수정에 대한 구체적인 의견을 위원장에게 제출할 수 있다. 심사위원의 이러한 의견이 있을 때는 위원장은 이러한 의견을 위원회의 명의로 저널 저자에게 통보하여 수정을 요구한다.

제 10 조(채택의 판정) : 기고된 논문의 채택여부는 위원회가 결정한다.

### 논 문 심 사 기 준

제 11 조(심사기준) : 위원이 개별적으로 위촉받은 논문을 심사하거나 위원회가 이의 채택여부를 판정함에 있어서 아래의 기준을 적용한다.

1. 연구 주제의 중요도(이론적, 실무적 및 교육적 기여도)
2. 연구 방법의 적절성과 창의성
3. 연구보고서의 의사전달 효과
4. 기타 편집기술상의 요건에 부합되는 여부

### 부 칙

제 12 조(발 효) : 본 규칙은 한국경영학회 1983년 2월 26일 전체이사회에서 출석인원 2/3 이상의 찬성으로 의결된 날로부터 시행한다.

제 13 조(개 정) : 본 규칙은 한국경영학회 이사회에서 출석인원 2/3 이상의 찬성으로 개정할 수 있다.

原 稿 募 集 案 內

1983年9月發刊豫定인「經營學研究」第13卷第1號에 研究論文, 研究노트, 博士學位論文抄錄, 書評 또는 이미 同誌에 掲載된論文에 대한 論評을 發表하시고 자 하는 會員께서는 그 原稿를 1983年 6月 30日까지 아래 住所로 보내주시 기 바랍니다.

서울特別市 麻浦區 新水洞 山1番地

西江大學校 會計研修院

高 廷 燮