Journal of the Korean Institute
of Industrial Engineers
Vol. 19, No.2, June., 1993.

Skip-k 저장방식†

Skip-k Storage Policy†

임상규*

Sang-gyu Lim*

Abstract

We address a new storage policy, called Skip-k storage, which gives storing priority to the items with relatively higher turnover rates.

Applying the policy in storage systems such as AS/RS will reduce expected travel distance considerably.

We analyze system behavior and find an appropriate skip size expressed by the system parameters in case arrival rates become larger enough.

1. 서 론

물류관리 분야 중에서도 저장이라는 기능을 갖는 창고나 Yard의 효율적 운영이 최근들어 중요한 요소가 되고 있다. 특히 이제는 자동창고의 도입이 본격화 되고 있는데, 그것이 高價의 저장시스템이라는 사실에 비하면 그 운영에 있어서는 아직까지 효율성이 많이 떨어지는 편이다.

동일하게 주어진 자동창고라도 활용하는 정도에 따라 저장 시스템의 throughput(단위시간당 저장 and/or 회수의 횟수)이나 평균적인 입, 출고시간(혹은 거리)에는 현저한 차이가 날 수 있으며 따라서 자동창고의 저장능력이나 저장효율을 증대시키기 위해서는 효율적인 운영방식이 적용되어야 할 것이다.

또한 자동창고의 하드웨어적인 설계문제도 그 창고의 저장공간들을 어떤 식으로 이용할 것인가 하는 문제와 연계될 때에만 최적의 설계가 도출될 수 있을 것이다.

자동창고의 운영에 관련이 있는 초기의 연구는 1960년대 후반에서부터 1970년대초반에 몇몇 학자 [1, 2, 5]들에 의해 수행되었는데 이들의 접근방식은 수리계획 모형을 이용한 것이었으며, 또한 자동창고 그 자체가 아닌 자동창고의 운영에 부분적으로 응용될 수 있는 것이었다.

그 이후에 Schwarz, Hausman, Graves[6, 7, 11]에 의해서 할당저장방법 등 高회전율 제품에 우선권을 주는 저장방법을 포함해서 자동

† 이 논문은 1991년도 교육부지원 한국학술진흥재단의 지방대학육성과제 학술연구 조성비에 의하여 연구되었음.

* 경상대학교 산업공학과

창고의 operation을 위한 기존의 저장방법들을 비교분석하는 등 꽤 본격적인 연구가 이루어졌다. 그들의 비교분석의 주 도구는 simulation이었다.

Bozer, White[4]는 자동창고에서 임의 저장방법을 적용했을 때의 system property를 분석하고 S/R(Storage/Retrieval) machine의 평균이동 거리를 이론적으로 구했다.

Coffman et al[4]과 Newell[10]은 COL 저장방식에 대한 이론적 연구를 수행하였는데, Newell[10]은 M/M/∞ 대기이론 모형을 설정하였고 그의 주 관심사는 제품들의 최대저장 활용도와 저장되지 않고 남아 있는 저장공간의 behavior를 연구하는 것이었다.

본 연구는 자동창고의 저장능력을 증대시키고 저장효율을 향상시키기 위한 저장방법, 즉 평균적인 입고/출고 時의 이동거리를 최소화 함으로써 입고/출고에 소요되는 시간을 줄이고 따라서 불필요한 재고시간을 대폭 줄일 수 있는 새로운 저장방식을 제시한다.

자동창고 운영에서의 주된 목적은 평균적인 저장(혹은 회수) 시간(혹은 거리)을 최소화하는 것인데, 본 연구에서는 입고과정을 arrival process로, 출고과정을 departure process로, AS/RS의 저장 공간들을 server로 보는 대기이론 모형을 설정하였다. 그리고, 입고, 저장, 출고 사이에서 일어나는 behavior를 분석하고 주요한 성질들을 도출하였다.

본 연구에서는 분석의 용이함을 위해 저장공간들이 일직선상에 있는 것(linear storage)으로 가정하고, 어떤 저장방식에서의 저장 품목들의 공간적 분포(spatial distribution)를 분석한다. (실제로 자동창고가 갖는 저장공간의 面, Tehebychev 이동을 가정하더라도, 저장시스템 내에 있는 품목들의 수는 linear storage의 경우와 똑 같으며 단지 저장공간들까지의 이동거리만 다를 뿐이다. 따라서 面으로 된 저장공간들을 이동거리에 따라 일련번호를 붙여 일직선 상에 펼쳐 놓은 것으로 생각하면 될 것이다).

또한 저장효율을 높일 수 있는 새로운 저장방식인 Skip-k 방식을 제시하고 skip size의 최적해를 근사적으로 구하였다.

2. 문제와 기본적 문형

어떤 저장시스템에 2가지 유형의 제품들을 저장한다고 생각하자. 하나의 제품류는 高회전율(h)유형이고 다른 하나는 低회전율(l)유형이다.(회전율이 높은 품목이라는 것은 그 품목이 저장되어 머무르는 시간이 상대적으로 짧다는 의미임.)

특별한 언급이 없는 한 본 연구는 다음의 가정들 하에서 이루어진 것이다.

- ○ 저장 rack은 충분히 많은 수의 저장공간들로 이루어져 있어서, 저장을 위해 창고에 도착하는 모든 품목들이 항상 어디엔가 저장된다.
- ○ 유일한 I/O point가 저장 rack의 한쪽 끝에 위치한다.
- ○ 모든 저장 공간들은 동일하고 n번째 저장공간은 I/O point에서 d_n단위 떨어져 있다($d_1 \leq d_2 \leq \cdots$).

본 연구에서는 선형저장(Linear Storage)인 경우, 즉 $d_n = n$인 경우를 주로 다룬다.

- ○ 언제나 어떤 하나의 저장공간에는 최대 1개의 품목이 저장될 수 있다.
- ○ single command 형식의 transaction만 일어난다.

그리고 다음의 대기모형을 설정한다.

$h[l]$ 유형 품목들이 각각 rate $\lambda_h[\lambda_l]$을 갖는 Poisson process를 따라 저장 rack에 도착한다고 가정하자. 또한 각각의 도착된 $h[l]$유형 품목은 각각 모수 $\mu_h[\mu_l]$을 갖는 지수분포를 따르는 시간만큼 저장되어 저장공간에 머무른다고 가정하자. 그러면, 이 저장 시스템에 있는 h와 l유형 품목들의 수(number)는 M/M/∞ 대기모형처럼 움직이게 될 것이다.

여기에서 우리의 목적은 평균적인 저장/회수 시간(거리)을 최소화 함으로써, 저장 시스템에서 품목을 저장하고 회수하는데 좀 더 적은 시

간을 쓸 수 있게끔 하는 것이다.

한편 우리의 대기모형이 갖는 중요한 성질을 하나 분석해 보면, stationary policy 下에서는 (우리가 고찰하고자 하는 저장방식들은 모두 stationary policy 들이다), 저장거리(storage distance)와 회수거리(retrieval distance)의 stationary 분포가 동일하다는 것이다. 특히 저장거리와 회수거리의 기대치가 동일한 값을 갖게 된다는 것이다.

이 성질은 기본적으로 PASTA(Poisson Arrivals See Time Averages)에 의해 증명될 수 있다[8].

따라서 우리는 저장거리이든 회수거리이든 분석에 용이한 하나를 선택하여 시스템의 분석에 접근할 수 있으며, 저장(혹은 회수)거리 대신에 이동거리(travel distance)라는 용어를 사용하여도 될 것이다.

3. 저장방식과 평균적 이동거리

어떤 저장방식을 적용하더라도 그 저장방식들이 stationary policy에 속한다면, 평균적인 이동거리를 어느 정도 이하로는 결코 줄일 수 없는 하한이 존재한다. 이에 대한 핵심적 아이디어는 하나의 이상적인 상황(실제로는 존재하지 않을지라도)을 설정하는 것이다.

즉 우리가 저장 시스템의 저장상황을 관찰했을때, 언제나 h유형 품목들이 I/O point에서 가까운 저장공간들을 꽉 채우고 있고 그 다음 가까운 저장공간들이 l유형 품목들로 꽉 채워져 있고 그다음 나머지 저장들은 비어 있는 상황만이 존재한다고 가정한다면 이러한 경우 이동거리의 기대치는 가장 적은 값으로 나타나게 될 것이다. 실제 저장시스템의 저장상황은 I/O에서 가까운 쪽부터 꽉 채워져 있지도 않을 것이며 (많은 시점에서 departure가 일어나므로) 또한 실제로는 h유형과 l유형 품목들이 서로 섞여서 있는 형태(물론 h 유형에 저장상의 우선권을 주는 저장방식이라면 대체적으로 h유형 품목들이 I/O에서 가까운 쪽에 많이 몰려 있기는 하겠지만)가 될 것이다.

이러한 하한을 수식으로 표현하면

$$D(\pi) \geq \frac{1}{\lambda} E\left[\frac{1}{E(Y^h_i)} \sum_{n=1}^{N^h} d_n + \frac{1}{E(Y^l_i)} \sum_{n=N^h+1}^{N^h+N^l} d_n\right] \quad \cdots\cdots (3.1)$$

여기에서

$D(\pi)$: policy π 下에서의 이동거리의 기대치

$Y^h_i[Y^l_i]$: i번째로 도착한 $h[l]$타입 item의 저장거리

$N^h(t)[N^l(t)]$: 시간 t에서 시스템에 있는 $h[l]$타입 item의 수

$\lambda = \lambda_h + \lambda_l$

따라서 선형저장 즉 $d_n = n$인 경우를 생각하면, 하한 D*는 다음과 같이 표현된다.

(3.1)식의 RHS

$$= \frac{1}{\lambda_h + \lambda_l} E[\mu_h \sum_{n=1}^{N^h} d_n + \mu_l \sum_{n=N^h+1}^{N^h+N^l} d_n]$$

$$= \frac{1}{\lambda_h + \lambda_l} E[\frac{\mu_h}{2} N^h(N^h+1) + \frac{\mu_l}{2} N^l(N^h+1+N^h+N^l)]$$

$$= 1 + P \cdot \frac{\rho_h}{2} + (1-P) \cdot (\rho_h + \frac{\rho_h}{2}) = D^* \quad \cdots\cdots (3.2)$$

여기에서 $P = \frac{\lambda_h}{\lambda_h + \lambda_l} = \frac{\lambda_h}{\lambda}$, $\rho_h = \frac{\lambda_h}{\mu_h}$, $\rho_l = \frac{\lambda_l}{\mu_l}$

한편 자동창고의 운영에서 이용되는 기본적인 저장방식에는 임의저장법(Randomized Storage : 임의의 빈 자리에 저장)과 COL저장법(Closest Open Location Storage : 가장 가까운 빈 자리에 저장)이 있다.

이러한 두 저장법은 나름대로 가치가 있고

또 실제 산업현장에서 많이 이용되고 있기는 하지만, 저장되는 제품의 회전율 특성에 의하여 품목들의 평균적인 이동거리를 증가시키는 "나쁜 상황"을 발생시킬 수 있다.

즉, 두가지 제품들을 구분없이 임의의 빈 자리나 가장 가까운 빈 자리에 저장하다보면, 회전율이 낮은 제품들(slow movers)이 I/O에서 상대적으로 가까운 저장공간들을 장기간 점유함에 따라 회전율을 높은 제품들(fast movers)이 상대적으로 자주 I/O에서 먼 저장공간들에 저장되게 되어, 결과적으로 평균적인 저장/회수 거리를 증가시키게 된다.

따라서 제품 종류들 사이의 회전율 차이가 현저히 존재하면 fast movers에게 저장상의 우선권(priority)을 갖게, 즉 fast movers가 slow movers보다 I/O point 가까이에 저장되는 기회가 많도록 하는 저장방식들이 검토되어야 한다.

이러한 저장방식들 중에서 대표적인 것으로는 할당저장법(Dedicated Storage)과 동급별 할당저장법(Class-Based Dedicated Storage) 등 [6, 7, 11]이 있는데, 동급별 할당저장법은 각 class사이에서 저장상의 overflow가 발생하지 않는다는 것을 가정한 셈인데(어떤 의미에서는 충분히 많은 저장공간을 각 class에 할당한 것인데), 실제로는 아무리 많은 저장공간을 할당하더라도 여전히 overflow의 확률은 존재하는 것이고, 또한 지나치게 많은 저장공간을 할당할 수도 없기 때문에 문제에 봉착하게 된다. 이러한 문제를 해결할 수 있는 새로운 저장방식으로는 Zone-z 저장방식 [9]이 있다.

Zone-z저장

- h유형 품목은 가장 가까운 빈자리(COL)에 저장된다.
- l유형 품목은 z번째 저장공간 너머 가장 가까운 빈자리에 저장된다.

(이때 zone size는 z이다).

Zone-z 저장방식에서 결정되어야 하는 것은 zone size인 z의 최적값(이동거리의 기대치를 최소화하는 z의 값)인데, 이 값을 closed form으로 구하기는 용이하지 않다. 따라서 아래에서 정의되는 근사적최적(asymptotically optimal)의 개념이 도입되었으며, 어떤 저장방식이 근사적 최적이면 그것이 저장효율이 높은 "좋은" 저장방식임이 밝혀졌다. 또한 zone size가 $[\lambda_h/\mu_h]$이면 근사적 최적이며, COL 저장방식은 근사적 최적이 아님이 증명되었다 [9]. 여기에서 $[\lambda_h/\mu_h]$는 λ_h/μ_h보다 작거나 같은 최대의 정수이다.

근사적 최적

만약 stationary policy π에서

$$\lim_{\lambda\to\infty}\frac{D(\pi)-D^*}{r(\lambda)}=0\text{이면} \tag{3.3}$$

policy π는 근사적 최적(asymptotically optimal)이다.

(위에서 $r(\lambda)$는 최적의 이동거리 기대치가 λ의 증가와 함께 증가한 비율(rate)을 나타내는데 우리가 다루는 선형저장에서는 $r(\lambda)=\lambda$이다)

4. Skip-k 저장

본 절에서는 고 회전율 제품(h유형 품목)에 저장상의 우선권을 주는 또 하나의 저장방식인 Skip-k 저장방법에 대해 연구한다.

Skip-k 저장

- h유형 품목은 가장 가까운 빈자리(COL)에 저장된다.
- l유형 품목은 (k+1)번째 빈자리에 저장된다. (이 때 skip size는 k이다).

$D_h(k)$와 $D^l(k)$를 각각 skip size가 k일때의 h유형과 l유형 품목들의 이동거리의 기대치라고 하면 skip-k 저장방식은 다음의 성질을 갖는다.

Property

$D^h(k)$는 k의 감소함수이고 $D^l(k)$는 k의 증가함수이다.

즉, $D^h(k)\geq D^h(k+1)$

$D^l(k) \leq D^l(k+1)$ (4.1)

여기에서 k=0, 1, 2, ….

위의 성질을 증명하기 위해서는, coupling argument를 이용하여, 다음의 관계가 성립함을 보여주면 될 것이다.

즉 동일한 확률공간에 Skip-k 저장방식과 Skip-(k+1)저장방식을 구성했을때, 모든 시점 t와 저장공간 m에 대해

$N^h(k+1 : m, t) \geq N^h(k : m, t)$

$N^l(k+1 : m, t) \leq N^l(k : m, t)$ (4.2)

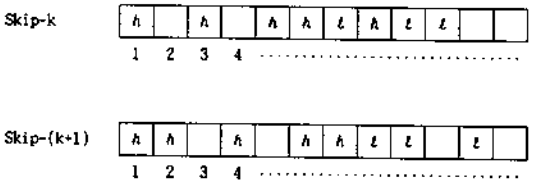
여기에서 $N^h(k : m, t)$와 $N^l(k : m, t)$는 각각 시점 t에서 skip size k일 때 첫 m 저장공간내에 있는 *h*유형과 *l*유형 품목의 수를 나타낸다.

(4.2)식은 다음과 같은 수학적 귀납법에 의해 증명될 수 있다. 먼저 t=0일 때 식(4.2)가 성립한다고 가정한 다음, 저장시스템의 저장상황에 변화를 가져오는 최초의 event시점(각 타입 item의 저장시점 및 회수시점의 4가지 경우)마다에서의 식(4.2)의 성립함을 보여주면 될 것이다.

(4.2)식은 도식적으로 설명하면, 다음과 같은 성질을 말해주고 있다.

동일한 두개의 선형저장공간을 갖고 있다고 가정하고 하나는 Skip-k 저장방식으로 운영하고 다른 하나는 Skip-(k+1) 저장방식으로 운영한다고 가정하자.

그러면 (4.2)식이 의미하는 바는 어느 시점에서나 *h*유형과 *l*유형 품목들의 공간분포(spatial distribution)가 예를 들면 다음의 형태를 갖게 된다는 것이다.



즉 어느 시점에서나, Skip-k에서의 i번째 *h* 유형 품목의 위치는 Skip-(k+1)에서의 i번째 *h*유형 품목의 위치보다 I/O에서 멀거나 기껏해야 같다는 것이다.

또한 *l*유형 품목에 대해서는, Skip-k에서의 i번째 *l*유형 품목의 위치는 Skip-(k+1)에서의 i번째 *h*유형 품목의 위치보다 I/O에서 가깝거나 최소한도 같다는 것이다.(이 사실은 simulation으로도 확인되었다).

또한 Skip-k 저장방식에서의 이동거리의 기대치는

$$D(k) = \frac{\lambda_h}{\lambda_h+\lambda_l} D^h(k) + \frac{\lambda_h}{\lambda_h+\lambda_l} D^l(k)$$

로 표현될 수 있기 때문에 D(k)를 최소로 하는 k의 값이 존재한다고 보아도 될 것이다. (더 정확히는 $D^h(k)$와 $D^l(k)$, 혹은 D(k)가 k에 대해 convex함수이어야 한다)

한편 skip size가 k인 Skip-k 저장시스템을 표현하는 difference equations를 다음과 같이 나타낼 수 있을 것이다. 먼저 첫 m개 까지의 저장공간들만 고려하고 m>k를 가정하자.(m≤k인 경우는 단지 *h*타입 품목만 저장되는 경우임).

그러면 $P_m(i, j)$를 m번째 저장공간까지에 있는 *h*유형과 *l*유형 품목들의 수에 대한 joint stationary distribution이라고 하면 평형 상태에서의 balance equation은 다음과 같다.

$$\{\lambda_h \cdot 1(i+j<m) + \lambda_l \cdot 1(i+j<m-k) + i\mu_h + j\mu_l\} P_m(i, j)$$
$$= P_m(i+1, j)\mu_h(i+1) \cdot 1(i+j+1 \leq m,\ j \leq m-k) + P_m(i, j+1)\mu_l(j+1) \cdot 1(i+j+1 \leq m,\ j+1 \leq m-k) + P_m(i-1, j)\lambda_h \cdot 1(i>0,\ i+j-1<m,\ j \leq m-k) + P_m(i, j-1)\lambda_l \cdot 1(j>0,\ i+j-1<m-k,\ j-1 \leq m-k) \quad (4.3)$$

여기에서

$$1(S) = \begin{cases} 1, & \text{state}(i, j) \subseteq S \\ 0, & \text{그 외에} \end{cases}$$

이제

$i+j=n$, $\gamma_n^m=\sum_{i=0}^{n} P_m(i, n-i)$이라고 두고,

(4.3)식의 좌변을 i에 대해 합하면

$$\sum_{i=0}^{n}\{\lambda_h \cdot 1(n<m)+\lambda_l \cdot 1(n<m-k)+i\mu_h$$

$$+(n-i)\mu_l\}P_m(i, n-i)=\gamma_n^m\lambda_l \cdot 1(n<m)+$$

$$\gamma_n^m\lambda_l \cdot 1(n<m-k)+\mu_h\sum_{i=0}^{n} iPm(i, n-1)+$$

$$\mu_l\sum_{i=0}^{n}(n-i)Pm(i, n-1)=\gamma_n^m\lambda_h \cdot 1(n<m)+$$

$$\gamma_n^m\lambda_l \cdot 1(n<m-k)+\gamma_n^m\{\mu_h E[N^h(m, t) \mid N$$

$$(m, t)=n]+\mu_l E[N^l(m, t) \mid N(m, t)=n]\} \quad (4.4)$$

여기에서

$$\alpha_n^m=\lambda_h \cdot 1(n<m)+\lambda_l \cdot 1(n<m-k) \quad (4.5)$$

$\beta_n^m=\mu_h E[N^h(m, t) \mid N(m, t)=n]+\mu_l$

$E[N^l(m, t) \mid N(m, t)=n]$라고 두면

식 (4.4)$=(\gamma_n^m \alpha_n^m \beta_n^m)$, $n=0\cdots m$.

반변에 식 (4.3)의 우변을 i에 대해 합하면

$$\mu_h\sum_{i=0}^{n}(i+1) \cdot 1(n<m)Pm(i+1, n-1)$$

$$+\mu_l\sum_{i=0}^{n}(n-i+1) \cdot 1(n<m)Pm(i, n-i+1)$$

$$+\lambda_h\sum_{i=1}^{n} 1(n<m+1)Pm(i-1, n-i)$$

$$+\lambda_l\sum_{i=0}^{n-1} 1(n<m-k+1)Pm(i, n-i-1)$$

$$=\gamma_{n+1}^m \beta_{n+1}^m+\gamma_{n-1}^m \alpha_{n-1}^m \quad n=1, \cdots, m-1.$$

양변을 같게 $\sum_{n=0}^{m}\sum_{i=0}^{n}$ LHS$=\sum_{n=0}^{m}\sum_{i=0}^{n}$ RHS 하면,

$$\gamma_{n+1}^m \beta_{n+1}^m+\gamma_{n-1}^m \alpha_{n-1}^m=\gamma_n^m(\alpha_n^m+\beta_n^m),$$

$$n=0, \cdots, m-1 \quad (4.6)$$

여기에서 $\gamma^m{}_{-1}\beta^m{}_0=\alpha^m{}_m=0$,

위의 difference equations을 풀면,

$$\gamma_n^m=\frac{\alpha_0^m\cdots\alpha_{l-1}^m}{\beta_l^m\cdots\beta_n^m}\gamma_0^m, \quad n=0, \cdots m$$

여기에서

$$\gamma_0^m=[1+\sum_{i=1}^{m}\frac{\alpha_0^m\cdots\alpha_{l-1}^m}{\beta_1^m\cdots\beta_l^m}]^{-1}$$

(단, $\sum_{n=0}^{m}\gamma_n^m=1$)

이제 $T_i^h(k), T_i^l(k)$를 각각 i번째 도착한 h 및 l유형 품목의 저장거리라고 하면

$$D^h(k)=\sum_{m=0}^{\infty} Pr\{T_i^h(k)>m\}$$

$$=\sum_{m=0}^{k} Pr\{T_i^h(k)>m\}+$$

$$\sum_{m=k+1}^{\infty} Pr\{T_i^h(k)>m\}$$

$$=\sum_{m=0}^{k}\frac{\rho_h^m/m!}{\sum_{i=0}^{m}\rho_h^i/i!}+\sum_{m=k+1}^{\infty}\gamma_m^m$$

여기에서 $\rho_h=\lambda_h/\mu_h$

$$D(k)=\sum_{m=0}^{\infty} Pr\{T_i^l(k)>m\}$$

$$=\sum_{m=0}^{k} Pr\{T_i^l(k)>m\}$$

$$+\sum_{m=k+1}^{\infty} Pr\{T_i^l(k)>m\}$$

$$=(k+1)+\sum_{m=k+1}^{\infty}\sum_{j=0}^{k}\gamma_{m-j}^{m}$$

그리고

$$D(k)=\frac{\lambda_h}{\lambda_h+\lambda_l}D^h(k)+\frac{\lambda_l}{\lambda_h+\lambda_l}D^l(k)$$

이제 평균적인 이동거리들을 구하기 위해서는 γ_n^m값이 구해져야 하고, 따라서 α_n^m과 β_n^m 값이 구해져야 한다. α_n^m은 식 (4.5)에 주어져 있으나, β_n^m은 값을 구하기가 어렵다.

그렇지만

$$\tilde{\beta}_n^m=\mu_h\frac{\rho_h}{\rho_h+\rho_l}+\mu_l\frac{\rho_l}{\rho_h+\rho_l}$$

명백히 $\tilde{\beta}_n^m<\beta^m{}_n<\mu_h$이다.

따라서 β_n^m 대신에 μ_h를 대입함으로서 $D^h(k)$와 $D^l(k)$ 각각에 대한 하한을 구할 수 있으며, β_n^m 대신에 $\tilde{\beta}_n^m$을 대입함으로써 상한을 구할 수 있다.

한편

$$\lim_{m\to\infty}\beta_n^m=\tilde{\beta}_n^m,$$

$$\lim_{m\to k}\beta_n^m=\mu_h$$

그래도 여전히 최적의 skip size를 결정하는 문제는 남아 있다. 그러나 Zone-z 저장방식과 Skip-k 저장방식이 갖는 밀접한 관련성, 특히 $\lambda=\lambda_h+\lambda_l$이 충분히 큰 값을 가질 때(heavy traffic), Zone-$[\rho_h]$ 저장방식이 근사적으로 최적이라는 사실을 이용하면 [9] 근사적으로 최적인 skip size를 다음과 같이 구할 수 있다.

Heavy traffic일 때, 만약 "Zone-$[\rho_h]$ 저장방식에서 l유형 품목들이 건너뛰는 빈 저장공간의 수 만큼"을 skip size로 하는 Skip-k 저장방식을 운용하면 이때의 Skip-k 저장방식과 Zone-$[\rho_h]$ 저장방식은 동일한 behavior를 보여주게 된다.

즉, N(m, t)를 시점 t에서, 첫 m번째 저장공간들 중에서 h유형이나 l유형에 의해 채워져 있는 저장공간의 수라고 하고,

$\hat{k}=[\rho_h]-E[N([\rho_h], t)]$로 놓으면

$\hat{k}$는 근사적을 최적인 skip size가 된다.

다시말하면, $\hat{k}$는 Zone-$[\rho_h]$ 저장방식이 적용되었을 때, 그 zone 내에 있는 빈 저장공간의 평균적인 갯수와 동일하다는 것이다.

한편 M/M/S/S (저장시스템의) 대기모형일 때 시스템 내에 있는 품목갯수의 stationary distribution은

$$P_n=\frac{s^n/n!}{\sum_{i=0}^{s}s^i/i!},\quad 0\le n\le s$$

[8]을 참고하면,

$$P_r\{overflow\}=[\sqrt{\frac{\pi}{2}}s+\frac{2}{3}+\frac{\sqrt{2\pi}}{24}s^{-1/2}$$

$$-\frac{4}{135}s^{-1}+0(s^{-2/3})]^{-1}$$

충분히 큰 s에 대해

$$\approx\sqrt{\frac{2}{\pi s}}$$

이제 M/M/$[\rho_h]$/$[\rho_h]$ 대기모형(s=$[\rho_h]$인

경우)을 생각하면,

$\hat{k}$=E [(0, [ρ_h])에서의 빈 저장공간 수]

$$=[\rho_h]-\sum_{n=0}^{[\rho_h]} nP_n$$

$$=[\rho_h]\cdot \Pr\{\text{overflow}\}$$

$$\approx\sqrt{\frac{2}{\pi}[\rho_h]}$$

Numerical example들을 표 1에 정리하였다.

표 1에서 D(0) 즉 skip size가 0인 경우는 COL 저장방식과 동일한 것임을 관찰할 수 있다.

또 마지막 예제인

λ_h=128, λ_l=32

μ_h=1.0, $\mu_l=\frac{1}{16}$인 경우를 생각하면

ρ_h=128이고, ρ_l=512이기 때문에 저장되어 있는 품목들 중 20%만이 h유형 품목이지만 이 20%를 차지하는 품목들의 활동(저장빈도)은 80%를 차지한다.

이러한 경우의 Skip-k 저장방식은 대단히 운영효율을 높이게 되는데 전산실험으로 나타난 결과는 최적의 skip size는 16이었으며 이때의 이동거리 기대치 D(16)=134이었다. 한편 근사적 최적인 skip size 10인 경우의 D(10)=144이었으며 COL 저장방식의 경우인 D(0)=300이었다.

표 1. Skip-k 저장방식에서의 평균적인 이동거리

모 수		D(0)	Simulation		근사적 최적	
			k*	D(k*)	$\hat{k}$	D($\hat{k}$)
μ_h=1.0 μ_l=0.4	λ_h= 32 λ_l= 8	28.1	6	24.9	5	24.9
	λ_h= 96 λ_l= 24	79.3	10	68.3	8	68.4
	λ_h=160 λ_l= 40	130.3	13	111.3	11	111.4
μ_h=1.0 μ_l=0.2	λ_h= 16 λ_l= 24	69.7	5	54.6	3	55.3
	λ_h= 48 λ_l= 72	202.7	8	153.3	6	153.5
	λ_h= 80 λ_l=120	333.8	9	250.0	7	250.5
μ_h=1.0 μ_l=0.4	λ_h= 16 λ_l= 24	40.0	3	35.5	3	35.5
	λ_h= 48 λ_l= 72	115.1	5	98.6	6	98.6
	λ_h= 80 λ_l=120	189.5	6	160.3	7	160.3
μ_h=1.0 μ_l=1/16	λ_h=128 λ_l= 32	300	16	134	10	144

따라서 우리는 근사적 최적인 Skip-10인 경우의 성능평가가 최적인 (simulation에 의한) Skip-16에 거의 가깝고, 또한 COL 저장방식에 비해서는 이동거리의 기대치를 52% 정도 줄일 수 있음을 알 수 있다.

5. 결 론

본 연구에서의 주된 관심사는 어떻게 하면 자동창고의 운영을 효율적으로 할 수 있겠는가 하는 것이었다. 이러한 문제를 해결하는 관건은 자동창고를 사용함에 있어서 효과적이고 적절한 저장방법을 개발, 적용시킬 수 있느냐에 달려 있다고 하겠다.

본 연구에서는 저장되는 제품들이 갖는 특성(특히 회전율의 차이)을 고려한 저장 방법을 개발하여 분석, 비교, 적용하여 보았다.

본 연구의 결과는 자동창고 뿐만 아니라 일반적인 창고나 yard의 저장, 주차장 운영 및 컴퓨터 memory allocation을 효율적으로 하는 데에 적용될 수 있을 것이다.

참 고 문 헌

1. Ballou, R.H., "Improving the Phsical Layout of Merchandise in Warehouses", Journal of Marketing, Volume 321, 60-34, 1967
2. Bazara, M.S., "Computerized Layout Design : A Branch and Bound Apporch, "AIIE Transaction, Volume 7, 432-438, 1975
3. Bozer, Y.A. and J.A. White, "Travel-Time Models for Automated Storage/Retrieval Systems", IIE Transactions, Volume 16, Numver 4, 329-338, 1984.
4. Coffman Jr. E.G., Kadota, T.T., and Shepp, L.A., "A Stochastic Model of Fragmentation in Dynamic Storage Allocation", SIAM J. Comput., Volume 14, 416-425, 1985.
5. Francis, R.L., "Sufficient Conditions for Some Optimum Property Facility Design", Operations Research, Volume 15, Number 3, 448-466, 1967.
6. Graves, S.C., Hasman, W.H., and Schwarz, L.B., "Storage-Retrieval Interleaving in Automatic Warehousing Systems", Management Science, Volume 23, Number 9, 935-945, 1977.
7. Hausman, W.H. Schwarz, L.B., and Graves, S.C., "Optimal Storage Assignment in Automatic warehousing Systems", Management Science, Volume 22, Number 6, 629-638, 1976
8. Lim, S., "Zoning in Storage Systems", Doctoral Dissertation, Georgia Institute of Technology., 1990.
9. Lim, S., "A Study on the Storage Systems for the Items Having Different Turnover Rates", Journal of the Korean Institute of Industrial Engineers, Volume 18, Number 1, 1-9, 1992.
10. Newll, G.F., The M/M/∞ Service System with Ranked Servers in Heavy Traffic, Springer-Verlag, 1984.
11. Schwarz, L.B., Graves, S.C., and Hausman, W.H., "Scheduling Policies for Automatic Warehousing Systems : Simulation Results", AIIE Transaction, Volume 10, Number 3, 260-270, 1978.