SCHOOL OF OPERATIONS RESEARCH
AND INDUSTRIAL ENGINEERING
COLLEGE OF ENGINEERING
CORNELL UNIVERSITY
ITHACA, NY 14853-3801

TECHNICAL REPORT NO. 949

December 1990

AN IMPROVED ALGORITHM FOR FINDING OPTIMAL LOT SIZING POLICIES FOR FINITE PRODUCTION RATE ASSEMBLY SYSTEMS

by

R.O. Roundy[1] and D. Sun[2]

[1]Research supported in part by NSF Grant DMC-8451984, and by AT&T Information Systems.

[2]Department of Commerce and Economics, Cariboo College.

An Improved Algorithm for Finding Optimal Lot Sizing Policies for Finite Production Rate Assembly Systems

R. O. Roundy
School of Operations Research and Industrial Engineering
College of Engineering, Cornell University

D. Sun
Department of Commerce and Economics
The University College of the Cariboo

Technical Report 949
College of Engineering, Cornell University

October 11, 1990

Revised June 25, 1991

Abstract

We show that an $O(n^3 \log n)$ algorithm can find optimal Power–of–Two Lot Size Policies for Finite Production Rate Assembly Systems. This improves an $O(n^5)$ algorithm proposed in Atkins, Queyranne and Sun's paper [1] (1992).

In their paper "Lot Sizing Policies for Finite Production Rate Assembly Systems" [1] (1992), Atkins, Queyranne and Sun provided an $O(n^5)$ algorithm to find optimal Power–of–Two Lot Size Policies for Finite Production Rate Assembly Systems. In this article we show that an $O(n^3 \log n)$ algorithm can solve the same problem. The organization of the paper is as follows. First, we rewrite the original relaxation problem (RP) in Atkins, Queyranne and Sun [1] (1992) to an equivalent problem (RP_1). Then, we present a mapping from this model to the model presented in Roundy [3] (1990). By using this mapping, we show an algorithm solving the original problem in $O(n^3 \log n)$. Finally, we give an example to illustrate the mapping procedure.

Refer to [1] (1992) for the notation, motivation, etc. We introduce the following equivalent formulation to the original relaxation problem of Atkins, Queyranne and Sun.

Lemma 1 (Equivalent Formulation)
Problem (RP)*:*

$$C^* = \min_Q f(Q) \triangleq \min_Q \sum_{i\in N}\left[\frac{K_i}{Q_i/\pi_0} + \sum_{j\in\langle i,1\rangle} H_{ij}\max_{\ell\in\langle i,j\rangle} Q_\ell\right]$$
$$\text{s.t.}\quad Q_i \geq 0 \qquad \forall\, i\in N$$

is equivalent to problem (RP_1)*:*

$$C_1^* = \min_q f_1(q) \triangleq \min_q \sum_{i\in N}\left[\frac{K_i}{q_{ii}/\pi_0} + \sum_{j\in\langle i,1\rangle} H_{ij}q_{ij}\right]$$
$$\text{s.t.}\quad q_{ij} \geq 0 \qquad \forall\, \langle i,j\rangle \in R, \tag{1a}$$
$$q_{ij} \leq q_{i,s(j)} \qquad \forall\, \langle i,s(j)\rangle \in R, \tag{1b}$$
$$q_{ij} \geq q_{s(i),j} \qquad \forall\, \langle s(i),j\rangle \in R, \tag{1c}$$

where R *is the set of all paths in* $G(N,A)$*.*

Proof. Suppose that $Q = (Q_1,\cdots,Q_n)$ is a feasible solution to (RP). Let

$$q_{ij} \triangleq \max_{\ell\in\langle i,j\rangle} Q_\ell, \qquad \forall\, \langle i,j\rangle \in R. \tag{2}$$

Then inequalities (1a), (1b) and (1c) hold, that is., $q = (q_{ij}|\langle i,j\rangle \in R)$ is also a feasible solution to (RP_1). Note also that $q_{ii} = Q_i, \forall\, i \in N$. Therefore, $f_1(q) = f(Q)$, and $C_1^* \leq C^*$.

Now suppose that q is a feasible solution to (RP_1). Let $Q_i = q_{ii}, \forall\, i\in N$. If $\ell \in \langle i,j\rangle \in R$, then by (1c) $q_{ij} \geq q_{s(i),j} \geq \cdots \geq q_{\ell j}$, and by (1b) $q_{\ell\ell} \leq q_{\ell,s(\ell)} \leq \cdots \leq q_{\ell j}$. Therefore, $Q_\ell = q_{\ell\ell} \leq q_{ij}$, $\forall\, \ell\in\langle i,j\rangle$, and $\max_{\ell\in\langle i,j\rangle} Q_\ell \leq q_{ij}$. Hence, $f(Q) \leq f_1(q)$, and $C^* \leq C_1^*$.

This implies $C^* = C_1^*$, i.e., problem (RP) is equivalent to problem (RP_1). □

The mapping is best described by defining three networks and by providing network-based reformulations of problem (RP_1). The three networks are defined as follows.

1. Network $G(N_1, A_1)$ corresponds directly to problem (RP_1).

$$
\begin{aligned}
N_1 &\triangleq \{\langle i,j\rangle \in R\}, \\
A_1 &\triangleq \{(\langle i,s(j)\rangle,\langle i,j\rangle)\,|\,\langle i,s(j)\rangle \in R\} \cup \{(\langle i,j\rangle,\langle s(i),j\rangle)\,|\,\langle s(i),j\rangle \in R\}
\end{aligned}
$$

For each node $\langle i,j\rangle$ in N_1, the setup cost and the holding cost are

$$
\begin{aligned}
K_{\langle i,j\rangle} &= \begin{cases} K_i\pi_0, & \text{if } i=j, \\ 0, & \text{if } i\neq j \end{cases} \\
H_{\langle i,j\rangle} &= H_{ij}.
\end{aligned}
$$

Note that problem (RP_1) can now be re-stated as

$$
\text{Problem } (RP_1^*): \quad \begin{cases} \min\limits_{q} & \sum\limits_{\langle i,j\rangle\in N_1} \left(\dfrac{K_{\langle i,j\rangle}}{q_{ij}} + H_{\langle i,j\rangle} q_{ij}\right) \\ \text{s.t.} & q_{ij} \geq q_{i'j'} \geq 0, \qquad\qquad \forall\, (\langle i,j\rangle,\langle i',j'\rangle) \in A_1. \end{cases}
$$

2. Let $D \triangleq \max\limits_{i_\ell\in L} |\langle i_\ell,1\rangle|$ be the length (the number of nodes in a route) of a longest leaf-route in $G(N,A)$, where L is the set of all leaves. Let $G(N',A')$ be the series system with

$$
\begin{aligned}
N' &\triangleq \{0,1,2,\cdots,D-1\} \\
A' &\triangleq \{(i,i-1)\,|\,i=1,2,\cdots,D-1\}.
\end{aligned}
$$

3. Let $G(N_2,A_2)$ be a graph defined by

$$
\begin{aligned}
N_2 &\triangleq \{\langle i,k\rangle\,|\,i\in N, k=0,1,2,\cdots,D-1\} \\
A_2 &\triangleq \{(\langle i,k\rangle,\langle s(i),k\rangle)\,|\,i\in N\backslash\{1\}, k=0,1,2,\cdots,D-1\} \\
&\quad \cup \{(\langle i,k\rangle,\langle i,k-1\rangle)\,|\,i\in N, k=1,2,\cdots,D-1\}
\end{aligned}
$$

The network $G(N_2,A_2)$ can be viewed as the cross product of $G(N',A')$ and of $G(N,A)$ (see Figures 1, 2 and 3). $G(N_2,A_2)$ has the structure that Roundy [3] requires. We embed $G(N_1,A_1)$ into $G(N_2,A_2)$ as follows.

$$
\text{node } \langle i,j\rangle \in N_1 \longmapsto \text{node } \langle i,k\rangle \in N_2, \qquad \text{with } k = D - |\langle j,1\rangle|.
$$

Costs for $G(N_2,A_2)$ are defined as follows.

$$
\begin{aligned}
K'_{\langle i,k\rangle} &= \begin{cases} K_{\langle i,j\rangle}, & \text{if } \langle i,j\rangle \in N_1 \longmapsto \langle i,k\rangle \in N_2, \qquad \text{with } k = D-|\langle j,1\rangle|, \\ 0, & \text{otherwise} \end{cases} \\
H'_{\langle i,k\rangle} &= \begin{cases} H_{\langle i,j\rangle}, & \text{if } \langle i,j\rangle \in N_1 \longmapsto \langle i,k\rangle \in N_2, \qquad \text{with } k = D-|\langle j,1\rangle|, \\ 0, & \text{otherwise} \end{cases}
\end{aligned}
$$

We now define Problem (RP_2) as

$$\text{Problem } (RP_2): \quad \begin{cases} \min\limits_{q'} & \sum\limits_{\langle i,j\rangle \in N_2} \left(\dfrac{K'_{\langle i,j\rangle}}{q'_{ij}} + H'_{\langle i,j\rangle} q'_{ij} \right) \\ \text{s.t.} & q'_{ij} \geq q'_{i'j'} \geq 0, \qquad \forall\, (\langle i,j\rangle, \langle i',j'\rangle) \in A_2. \end{cases} \qquad 24;1H$$

The costs for $G(N_2, A_2)$ are obviously selected to make problems (RP_1^*) and (RP_2) equivalent. Let $S = \{\langle i,k\rangle \in N_2 : k \leq D - 1 - |\langle i,1\rangle|\}$. Note that nodes in S have no corresponding nodes in N_1. The setup costs and holding costs corresponding to these nodes are zero, and no arc in A_2 goes from a node in S to a node in $N_2 \backslash S$. Using these facts the equivalence between problems (RP_1^*) and (RP_2) is easily verified.

The algorithm for solving (RP) can be summarized as follows.

1. Construct $G(N_2, A_2)$ as described above

2. Use the algorithm suggested by Roundy [3] to solve Problem (RP_2) over network $G(N_2, A_2)$ in the time of $O(|N_2| D \log |N_2|)$. Note that $|N_2| \leq n^2$ and $D \leq n$, so $O(|N_2| D \log |N_2|) \leq O(n^3 \log n)$. Let the solution to Problem (RP_2) be $q'_{\langle i,k\rangle}$ for every node $\langle i,k\rangle \in N_2$.

3. In order to get the solution to the relaxation problem (RP) over the network $G(N_1, A_1)$, we use the inverse mapping from $G(N_2, A_2)$ to $G(N_1, A_1)$:

   $$\langle i,k\rangle \in N_2 \quad \longmapsto \quad \begin{cases} \emptyset, & \text{if} \quad |\langle i,1\rangle| \leq D - 1 - k \\ \langle i,j\rangle \in N_1, & \text{if} \quad \langle j,1\rangle \subseteq \langle i,1\rangle \in G(N,A) \text{ such that } |\langle j,1\rangle| = D - k \end{cases}$$

   The solution to problem (RP_1) over the network $G(N_1, A_1)$ is:

   $$q_{ij} = q_{\langle i,j\rangle} = q'_{\langle i,k\rangle}, \qquad \text{if } \langle i,k\rangle \in N_2 \longmapsto \langle i,j\rangle \in N_1.$$

   When carefully implemented, the run time for this step is $O(|N_2|) \leq O(n^2)$.

4. Let $Q_i \stackrel{\triangle}{=} q_{ii}$, $\forall\, i \in N$. We have the solution to the original problem (RP).

5. Using the optimal rounding method in Roundy [2] (1983) to derive, in $O(n \log n)$ time, an optimal power–of–two lot size policy for the finite production rate assembly systems with effectiveness at least 98%.

It is easy to see that the total time to solve the problem is bounded by the time to solve the relaxation problem over network $G(N_2, A_2)$ in step 2, which is $O(n^3 \log n)$.

The following example illustrates the mapping process.

Example. The following example of an assembly system $G(N, A)$ in Figure 1 with 7 facilities illustrates the embedding procedure. The length of the longest leaf–route, which corresponds to the series network $G(N', A')$ in Figure 2, is four, i.e., $D = 4$. Graph $G(N_2, A_2)$ in Figure 3 is the network corresponding to problem (RP_2). It is the Cartesian product of graph $G(N, A)$ and graph $G(N', A')$. The graph $G(N_1, A_1)$ in Figure 4 is imbedded in graph $G(N_2, A_2)$.

It is easy to verify the mapping from $G(N_1, A_1)$ to $G(N_2, A_2)$. The following examples illustrate the inverse mapping from $G(N_2, A_2)$ to $G(N_1, A_1)$:

If $\langle i, k\rangle = \langle 7, 3\rangle \in N_2$, then $i = 7$, $k = 3$ and $D - k - 1 = 4 - 3 - 1 = 0 < 4 = |\langle 7, 1\rangle| = |\langle i, 1\rangle|$, so there is a corresponding node in $G(N_2, A_2)$. Therefore, $|\langle j, 1\rangle| = D - k = 4 - 3 = 1$ and $j = 1$, i.e., $\langle 7, 3\rangle \in N_2 \longmapsto \langle 7, 1\rangle \in N_1$.

If $\langle i, k\rangle = \langle 7, 0\rangle \in N_2$, then $i = 7$, $k = 0$ and $D - k - 1 = 4 - 0 - 1 = 3 < 4 = |\langle 7, 1\rangle| = |\langle i, 1\rangle|$. Therefore, $|\langle j, 1\rangle| = D - k = 4 - 0 = 4$ and $j = 7$, i.e., $\langle 7, 0\rangle \in N_2 \longmapsto \langle 7, 7\rangle \in N_1$.

If $\langle i, k\rangle = \langle 6, 3\rangle \in N_2$, then $i = 6$, $k = 3$ and $D - k - 1 = 4 - 3 - 1 = 0 < 3 = |\langle 6, 1\rangle| = |\langle i, 1\rangle|$. Therefore, $|\langle j, 1\rangle| = D - k = 4 - 3 = 1$ and $j = 1$, i.e., $\langle 6, 3\rangle \in N_2 \longmapsto \langle 6, 1\rangle \in N_1$.

If $\langle i, k\rangle = \langle 6, 0\rangle \in N_2$, then $i = 6$, $k = 0$ and $D - k - 1 = 4 - 0 - 1 = 3 = |\langle 6, 1\rangle| = |\langle i, 1\rangle|$. Therefore, $\langle 6, 0\rangle \in N_2 \longmapsto \emptyset \in N_1$.

The following table summarizes the mapping between $G(N_1, A_1)$ and $G(N_2, A_2)$.

$\langle i,j\rangle \in N_1 \Longleftrightarrow \langle i,k\rangle \in N_2$	
$\langle 7,1\rangle$	$\langle 7,3\rangle$
$\langle 7,3\rangle$	$\langle 7,2\rangle$
$\langle 7,5\rangle$	$\langle 7,1\rangle$
$\langle 7,7\rangle$	$\langle 7,0\rangle$
$\langle 6,1\rangle$	$\langle 6,3\rangle$
$\langle 6,3\rangle$	$\langle 6,2\rangle$
$\langle 6,6\rangle$	$\langle 6,1\rangle$
$\langle 5,1\rangle$	$\langle 5,3\rangle$
$\langle 5,3\rangle$	$\langle 5,2\rangle$

$\langle i,j\rangle \in N_1 \Longleftrightarrow \langle i,k\rangle \in N_2$	
$\langle 5,5\rangle$	$\langle 5,1\rangle$
$\langle 4,1\rangle$	$\langle 4,3\rangle$
$\langle 4,2\rangle$	$\langle 4,2\rangle$
$\langle 4,4\rangle$	$\langle 4,1\rangle$
$\langle 3,1\rangle$	$\langle 3,3\rangle$
$\langle 3,3\rangle$	$\langle 3,2\rangle$
$\langle 2,1\rangle$	$\langle 2,3\rangle$
$\langle 2,2\rangle$	$\langle 2,2\rangle$
$\langle 1,1\rangle$	$\langle 1,3\rangle$

Figure 1: Graph $G(N, A)$

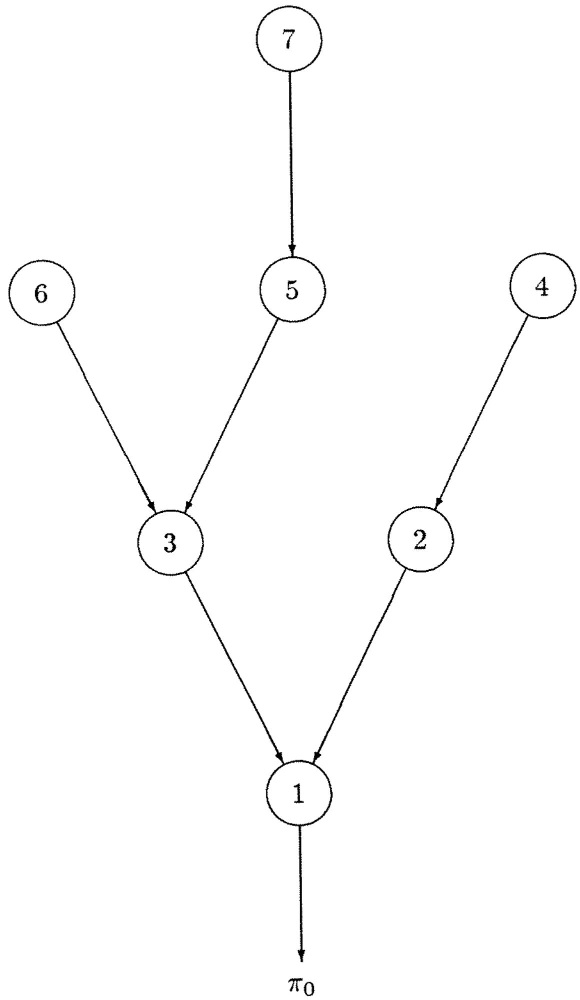


Figure 2: Graph $G(N', A')$

Figure 3: Graph $G(N_2, A_2) = G(N, A) \times G(N', A')$

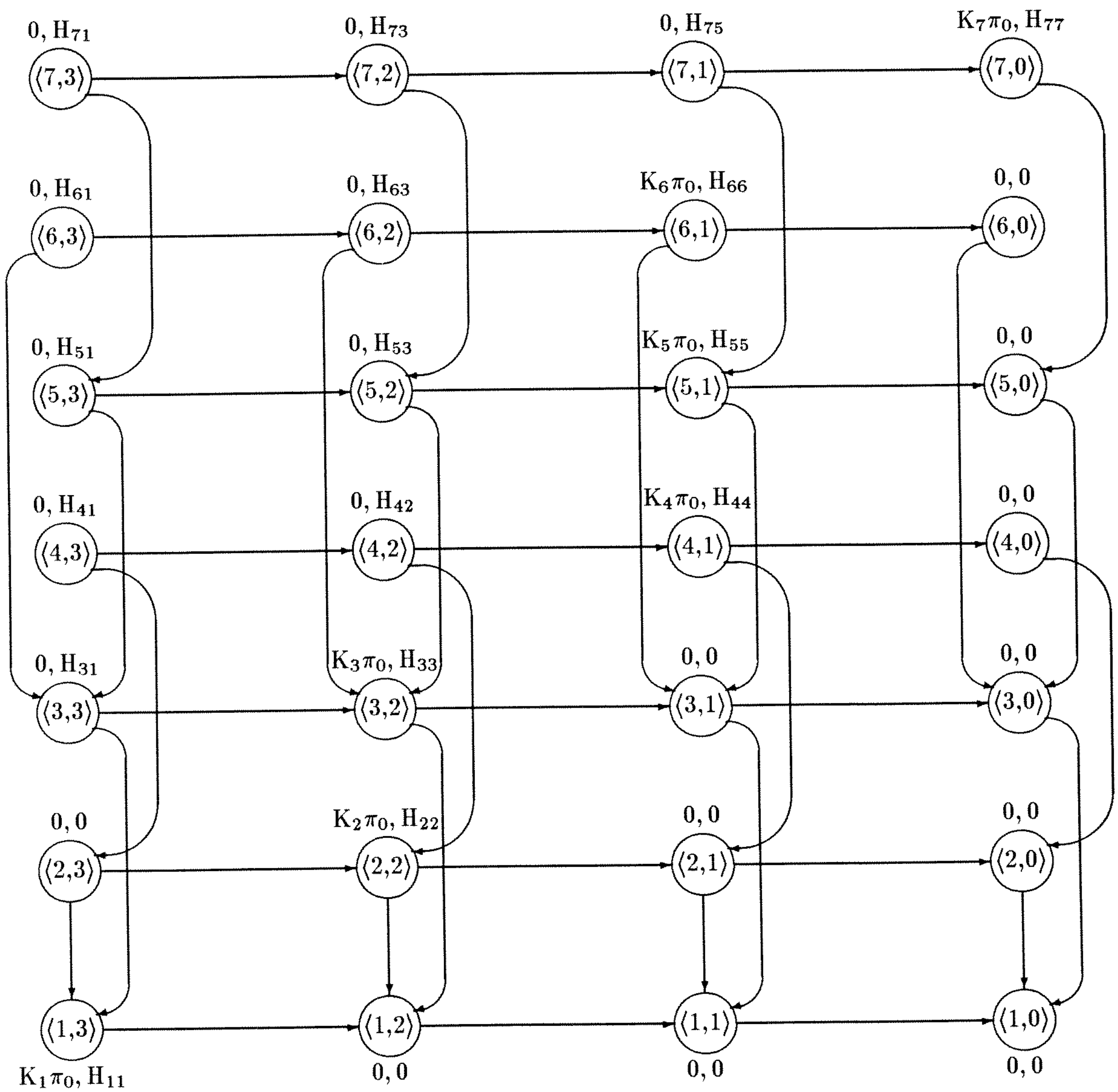
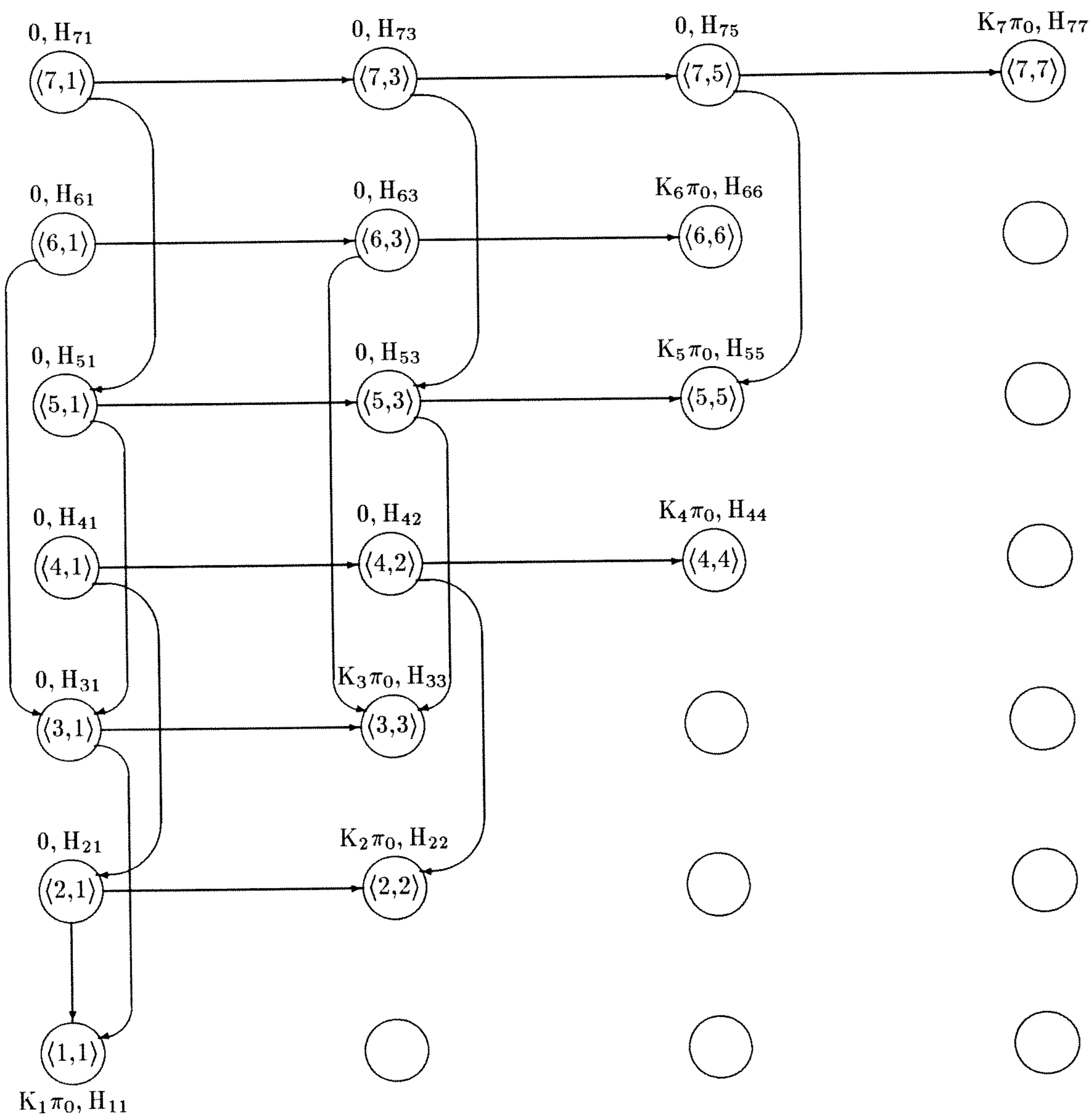


Figure 4: Graph $G(N_1, A_1)$, which is embedded in graph $G(N_2, A_2)$



References

[1] D. Atkins, M. Queyranne, and D. Sun. "Lot Sizing Policies for Finite Production Rate Assembly Systems". *Operations Research*, January–February 1992, Vol.40, No.1, pp.126–141.

[2] R. O. Roundy. "94%–Effective Lot–Sizing in Multi–Stage Assembly Systems". Technical Report No. 674, School of Operations Research and Industrial Engineering College of Engineering Cornell University, Ithaca, New York 14850, September 1983.

[3] R. O. Roundy. "Computing Nested Reorder Intervals for Multi–Item Distribution Systems". *Operations Research*, January–February 1990, Vol.38, No.1, pp.37–52.