ELSEVIER

European Journal of Operational Research 120 (2000) 511–524

EUROPEAN JOURNAL OF OPERATIONAL RESEARCH

www.elsevier.com/locate/orms

Theory and Methodology

Optimal multiple stage expansion of competence set

Jian-Ming Li [a], Chin-I Chiang [b], Po-Lung Yu [c,*]

[a] Institute of Applied Mathematics, Guizhou University of Technology, Guiyang, Guizhou Province, People's Republic of China
[b] Institute of Traffic and Transportation, National Chiao Tung University, Taipei, Taiwan, ROC
[c] School of Business, University of Kansas, Lawrence, KS 66045, USA

Received 14 April 1998; accepted 8 September 1998

Abstract

In this paper, we consider the expansion processes of competence sets which have asymmetric cost functions, intermediate skills, and compound skills; among the skills, cyclic connections are possible. We introduce the concept of the stage expansion process (SEP) of the competence set, and provide mathematical programming methods to find a minimal cost SEP and the ordering of expansion. © 2000 Elsevier Science B.V. All rights reserved.

Keywords: Decision theory; Competence set analysis; Stage expansion process; Expansion process; Habitual domains

1. Introduction

To be a medical doctor, teacher, certified accountant, stock broker, electrician, computer programmer, etc., one must go through a sequence of training and/or examinations in order to gain the basic skills to do a good job.

Indeed, for each significant decision problem, there is a competence set consisting of ideas, knowledge, information, and skills for its satisfactory solution. When the decision maker (DM) thinks that he/she has acquired the needed competence set as perceived, he/she will be confident in making the decision. Otherwise, the DM may want to expand the competence set. In order to attain adequate competence sets, billions of dollars have been spent by corporations and individuals in job training, and many successful people spend many hours a week in active learning. Society, in order to certify the quality of work, issues certificates, diplomas and licenses to the people who have acquired certain competence sets.

Competence set analysis was first introduced by Yu [1] as an application part of habitual domains [1,2]. Its mathematical foundation was reported in Yu and Zhang [3]. Mathematical methods to attain more efficient ways to acquire the needed competence sets under various assumptions have been subsequently reported in [3–8].

* Corresponding author. Tel.: +1 785 864 7565; fax: +1 785 864 5328.

PII: S0377-2217(98)00397-X

More specifically, when the cost function $c(x, y)$, acquiring y from x, is symmetric (i.e., $c(x, y) = c(y, x)$), Yu and Zhang [3,7,8] use the next-best method to acquire the optimal expansion process. When the cost function is asymmetric and the digraph (directed graph), which describes the interconnections of expansion among the skills, contains cycles, Shi and Yu [6] introduce a concept of *tree expansion process* and an integer programming method to get an optimal expansion process. Note, in the expansion process, there are collections of individual skills known as *compound skills* (see Definition 2.1 of this paper) which can facilitate the acquisition of other skills; and there are some skills which are not really needed for solving the problem but can facilitate the acquisition of the needed skills. The latter are known as *intermediate skills*. Li and Yu [5] use the concept of *deduction graph* and develop an integer programming method to solve the problem of competence set expansion in which compound skills and intermediate skills exist in the digraph which contains no cycles. Recently, Li [4] uses the concept of the deduction graph and proposes an integer programming method to solve the problem of competence set expansion in which compound skills, intermediate skills, and cycles exist in the digraph, but he does not give the proof to his method.

In this paper, we focus on how most efficiently to help the DM acquire the needed competence set under very general conditions. We introduce a new concept of the stage expansion process (SEP) to study competence set expansion problems that have compound skills, intermediate skills and cycles in the digraph. A major significance of using this concept of the SEP is that we can apply a forward method to derive the fundamental theorem (Theorem 3.1). This forward method is intuitively clear, and much easier to understand than that of the backward method of [6]. In addition, the forward method allows us to study the expansion process for digraphs which contain compound and intermediate nodes, which was not included in [6]. We show that the cost of the minimal cost expansion process is equal to that of the minimal cost SEP. Thus to find an optimal expansion process, we only need to find an optimal SEP. An integer programming method is developed to acquire the minimal cost SEP, which can simultaneously determine the ordering of stages for the skills to be acquired in the expansion process. This makes it easier for us to depict graphically the expansion process of the competence set.

The paper is organized as follows. In Section 2, we describe some preliminary concepts of expansion processes when there are compound skills, intermediate skills and cycles. In Section 3, we introduce the new concept of the SEP and provide three important theorems. In Section 4, we first prove that the total cost of the minimal cost expansion process is equal to that of the minimal cost SEP, then we develop integer programming methods to find a minimal cost SEP. Finally, we give an example to illustrate the methods.

2. Expansion process

As we know, if we have already acquired the skills of geometry and algebra, then it is easier for us to learn calculus than if we learned geometry or algebra alone. Geometry and algebra together are called a compound skill, while geometry or algebra alone is called a (precedent) singleton skill. Each skill may be represented by a node in a directed graph. The directed arcs in the graph indicate the sequence for the skills to be obtained.

Let $\mathrm{HD} = \mathrm{Sk} \cup I \cup C \cup \mathrm{Tr}$, where HD (habitual domains) is the *set of skills related to solving a particular problem*, Sk is the *already acquired competence set* (Skill), Tr is the *true competence set* needed for solving the problem, I is the set of all *intermediate skills*, and C is the set of all *compound skills*.

Let G be the digraph formed by all nodes or skills in HD and their directed arcs *including* the arcs among the compound nodes and their singleton nodes. Denote by $\underline{G}$ the digraph formed by all nodes or skills in HD and their directed arcs *excluding* the arcs among the compound nodes and their precedent nodes.

Let U and $\underline{U}$ be the collections of arcs in G and $\underline{G}$, respectively. That is:

$U = \{(x, y) | (x, y) \text{ is an arc of } G\}$,

$\underline{U} = \{(x, y) | (x, y) \text{ is an arc of } \underline{G}\}$.

Let $B(x)$ and $A(x)$ be the nodes immediately *before* and *after* x, respectively, with respect to U; while $\underline{A}(x)$ is that after x with respect to $\underline{U}$. That is:

$B(x) = \{y | (y, x) \in U\}$,

$A(x) = \{y \in \text{HD} \backslash \text{Sk} | (x, y) \in U\}$,

$\underline{A}(x) = \{y \in \text{HD} \backslash \text{Sk} | (x, y) \in \underline{U}\}$.

Note, a compound skill is obtained only when all of its precedent singleton skills have been obtained. In order to make compound skills or nodes precise, following Definition 2.1 of [5], we have

Definition 2.1. A node x_c in G is a compound node if x_c can be decomposed into a set of singleton nodes $\{x_{c1}, \ldots, x_{ck}\}$ and $B(x_c) = \{x_{c1}, \ldots, x_{ck}\}$. Denote $x_c = x_{c1} \wedge x_{c2} \wedge \ldots \wedge x_{ck}$.

Remark 2.1. There is no cost to acquire a compound node x_c from $x_{ci} \in B(x_c)$; however, if we want to get x_c, we must have acquired all x_{ci} first. Also by Definition 2.1, for every compound node x_c, except for x_{ci} $(i = 1, 2, \ldots, k)$, there is no other node connecting to x_c . In the following, by connection between x_{ci} and x_c (as a directed arc), we mean that the connection is from x_{ci} to x_c.

Let $c(x, y)$ be the cost of acquiring y from x.

Definition 2.2. We call sequence $\psi = (x_{k1}, x_{k2}, \ldots, x_{kn})$, with $\{\psi\} = \{x_{k1}, x_{k2}, \ldots, x_{kn}\} \subset \text{HD} \backslash \text{Sk}$ an expansion process from Sk to Tr, if

(a) $\text{Tr} \backslash \text{Sk} \subset \{\psi\}$,

(b) for $x_{ki} \in C, B(x_{ki}) \subset \text{Sk}_{i-1}$; for $x_{ki} \notin C, B(x_{ki}) \cap \text{Sk}_{i-1} \neq \emptyset$, where $\text{Sk}_0(\psi) = \text{Sk}$, $\text{Sk}_i(\psi) = \text{Sk}_{i-1}(\psi) \cup \{x_{ki}\}, i = 1, \ldots, n$.

We call $\text{Sk}_i(\psi)$ the ith *step* of the expansion process ψ, $C_i(\psi) = \min\{c(x, x_{ki}) \mid x \in \text{Sk}_{i-1}(\psi)\}$ the ith *step expansion cost*, $1 \leqslant i \leqslant n$; and $C(\psi) = C_1(\psi) + \cdots + C_n(\psi)$ the *total cost of the expansion process* ψ.

We call ψ^* a *minimal cost expansion process* if $C(\psi^*) = \min\{C(\psi) \mid \psi \text{ is an expansion process}\}$. In order to simplify notation, when the context is clear we will write $\{\psi\}$ simply by ψ. The following example is modified from Example 2.1 of [5].

Example 2.1. Let $\text{HD} = \{a, b, c, a \wedge b, D, e, f, e \wedge f, g, h\}$, $\text{Sk} = \{a, b, c, a \wedge b\}$, $\text{Tr} = \{d, f, g, h\}$, $I = \{e\}$, $C = \{a \wedge b, e \wedge f\}$. The cost of acquiring one skill (in the first row) from another skill (in the first column) is given in Table 1, the empty cells indicate that it is practically impossible to acquire one skill (in the first row) from its corresponding skill in the table. How do we efficiently get the Tr?

Note, $\psi = (f, d, e, e \wedge f, g, h)$ is an expansion process. Here, $\text{Sk}_0 = \{a, b, c, a \wedge b\}$, $\text{Sk}_1 = \{a, b, c, a \wedge b, f\}$, $\text{Sk}_2 = \{a, b, c, a \wedge b, f, d\}$, $\text{Sk}_3 = \{a, b, c, a \wedge b, f, d, e\}$, $\text{Sk}_4 = \{a, b, c, a \wedge b, f, d, e, e \wedge f\}$, $\text{Sk}_5 = \{a, b, c, a \wedge b, f, d, e, e \wedge f, g\}$, $\text{Sk}_6 = \{a, b, c, a \wedge b, f, d, e, e \wedge f, g, h\}$, $C(\psi) = 2.5 + 1 + 1 + 1.5 + 1.5 = 7.5$.

It will be seen that ψ is not a minimal cost expansion process. *Our problem is to get a minimal cost expansion process efficiently*.

Table 1
Costs required for new skills

	d	e	f	g	h
a	1		3	4	
b	2			4	
c	2	3			
$a \wedge b$	1		2.5	3.5	
d		1	1.8		
e				4	4
f	2	2		2	2
$e \wedge f$	2			1.5	1.5

3. Stage expansion process

When Sk contains only one skill and there is no compound skill and intermediate skill in G, Shi and Yu [6] introduced the concepts of the tree construction sequence and the tree expansion process to acquire the minimal tree expansion process, which is also a minimal cost expansion process. But with compound skills in G, it is impossible for us to use the concepts of the tree construction sequence and the tree expansion process again. How can we find a minimal expansion process in this situation? In order to solve this problem, we introduce the concept of the SEP.

Let V ($V \subset U$) be a set of arcs. Define:

$$B_v(x) = \{y \in \mathrm{HD} | (y,x) \in V\},$$
$$A_v(x) = \{y \in \mathrm{HD} \backslash \mathrm{Sk} | (x,y) \in V\},$$
$$\underline{A}_v(x) = \{y \in \mathrm{HD} \backslash \mathrm{Sk} | (x,y) \in V \cap U\}.$$

In the following, given $M_j = (x_1^j, \ldots, x_{nj}^j)$, in order to simplify the notation, the set $\{M_j\} = \{x_1^j, \ldots, x_{nj}^j\}$, without confusion, will be written simply as M_j.

Definition 3.1. Let $V \subset U$, $M_j = (x_1^j, \ldots, x_{nj}^j) \subset \mathrm{HD} \backslash \mathrm{Sk}$, $j = 1, 2, \ldots, r$. We call the sequence $\psi = (M_1, M_2, \ldots, M_r)$ with respect to V SEP, if it satisfies:

(i) $\mathrm{Tr} \backslash \mathrm{Sk} \subset \{\psi\} = \cup\{M_j | j = 1, \ldots, r\}$;

(ii) when $x_i^1 \in M_1 \backslash C$, $\exists$ $x \in \mathrm{Sk}$, such that $B_v(x_i^1) = \{x\}$; when $x_i^1 \in M_1 \cap C$, $B(x_i^1) \cap M_1 \neq \emptyset$, $B(x_i^1) \subset \mathrm{Sk} \cup \{x_1^1, \ldots, x_{i-1}^1\}$;

(iii) for $j = 2, \ldots, r$, when $x_i^j \in M_j \backslash C$, $\exists$ $y \in M_{j-1}$, such that $B_v(x_i^j) = \{y\}$; when $x_i^j \in M_j \cap C$, $B(x_i^j) \cap M_j \neq \emptyset$, $B(x_i^j) \subset \mathrm{Sk} \cup M_1 \cup \cdots \cup M_{j-1} \cup \{x_1^j, \ldots, x_{i-1}^j\}$.

Remark 3.1. If $\psi = (M_1, M_2, \ldots, M_r)$ with respect to V is an SEP, then we call M_j the jth stage expansion set, j is the ordering number of stage M_j, and there is no connection between any two singleton nodes or skills of M_j. If x_c is a compound node or skill of M_j, then it has no successor node in M_j. For each compound node $x_c = x_{c1} \wedge x_{c2} \wedge \cdots \wedge x_{ck}$, if $t = \max\{j | x_{ci} \in M_j, i = 1, \ldots, k\}$, then $x_c \in M_t$. This arrangement is based on the idea that if we get all the $x_{ci} (i = 1, \ldots, k)$, then we automatically get x_c. In other words, if x_{ct} is the last skill we get among all the skills of $\{x_{c1}, x_{c2}, \ldots, x_{ck}\}$, when we get x_{ct}, we automatically get x_c. Thus x_{ct} and x_c are acquired in the same stage. Stage expansion depicts the ordering of the skills acquired.

Example 3.1. From Example 2.1, $\psi = (f, d, e, e \wedge f, g, h)$ with respect to $V = \{(a \wedge b, f), (a \wedge b, d), (d, e), (e \wedge f, g), (e \wedge f, h)\}$ is a SEP (Fig. 1). Note, $M_1 = \{f, d\}$, $M_2 = \{e, e \wedge f\}$, $M_3 = \{g, h\}$. Let $V_1 = \{(a \wedge b, f), (a \wedge b, d), (f, e), (e \wedge f, g), (e \wedge f, h)\}$. We see that ψ with respect to V_1 is also an SEP (Fig. 2), with $M_1 =$

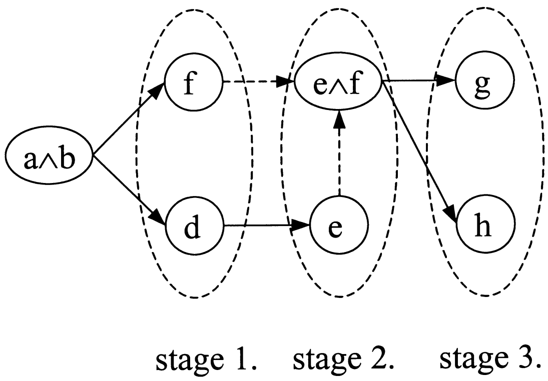

Fig. 1. The first SEP.

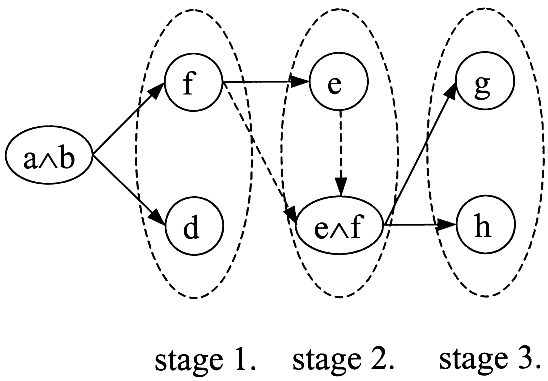

Fig. 2. The second SEP.

$\{f, d\}$, $M_2 = \{e, e \wedge f\}$, $M_3 = \{g, h\}$. Although these two SEPs have the same expansion stages, they are different SEPs.

Assumption 3.1. Let X be a set of skills ($X \subset$ HD\Sk), V a set of arcs ($V \subset U$). In the following, by "some set of skills X and some set of arcs V", we implicitly assume that V is the set of arcs consisting of some directed connections among the skills in $X^* = X \cup \{x \in \mathrm{Sk} | A_v(x) \cap X \neq \emptyset\}$. When compound node $x_c \in X$ and $x_{ci} \in B(x_c) \cap X^*$, for simplicity, we assume $(x_{ci}, x_c) \in V$.

Theorem 3.1. *Let X be a set of skills* ($X \subset$ HD\Sk), *V a set of arcs* ($V \subset U$), $X^* = X \cup \{x \in \mathrm{Sk} | A_v(x) \cap X \neq \emptyset\}$; *$X$ and V form an SEP if and only if*:

(i) Tr\Sk $\subset X$;
(ii) *for each* $x \in X \backslash C, B_v(x)$ *is a singleton and* $B_v(x) \subset X^*$;
(iii) *for each* $x \in X \cap C$, $B(x) \subset X^*$;
(iv) *there is no sequence* $(x_{p1}, x_{p2}, \ldots, x_{pk}) \subset X$ *satisfying*

$$x_{p1} \in B_v(x_{p2}), \ldots, x_{pk-1} \in B_v(x_{pk}), x_{pk} \in B_v(x_{p1}). \tag{1}$$

Proof. For necessity. If X and V form an SEP, by Definition 3.1, (i)–(iii) hold obviously. If there is a sequence $(x_{p1}, x_{p2}, \ldots, x_{pk}) \subset X$ satisfying (1), then from $x_{pk} \in B_v(x_{p1})$, x_{pk} is before x_{p1} in the SEP, but from

$x_{p1} \in B_v(x_{p2}), \ldots, x_{pk-1} \in B_v(x_{pk})$, we know that x_{p1} is before x_{pk} in the SEP, which leads to a contradiction. Thus (iv) holds also.

For sufficiency. Write:

$$X_0 = X,$$
$$M_1^s = \{x \in X_0 | B_v(x) \subset \text{Sk}, \text{if } x \notin C\},$$
$$M_1^c = \{x \in X_0 | B(x) \subset \text{Sk} \cup M_1^s \text{ and } B(x) \cap M_1^s \neq \emptyset, \text{if } x \in C\},$$
$$M_1 = M_1^s \cup M_1^c,$$
$$X_1 = X_0 \backslash M_1,$$
....
$$M_i^s = \{x \in X_{i-1} | B_v(x) \subset M_{i-1},\ x \notin C\},$$
$$M_i^c = \{x \in X_{i-1} | B(x) \subset \text{Sk} \cup M_1 \cup \ldots \cup M_{i-1} \cup M_i^s \text{ and } B(x) \cap M_i^s \neq \emptyset,\ x \in C\},$$
$$M_i = M_i^s \cup M_i^c,$$
$$X_i = X_{i-1} \backslash M_i,\ \ i = 1, 2, \ldots, n.$$

From the above definition, we see:

$$X_i = X_0 \Big\backslash \bigcup_{1 \leq j \leq i} M_j, M_i \cap M_j = \emptyset, \quad \text{with } i \neq j,\ X = \bigcup_{1 \leq j \leq n} M_j.$$

Note that if $X_{i-1} \neq \emptyset$, then $M_i \neq \emptyset$. Otherwise, for every $x \in X_{i-1}$, let us consider two possible cases: Case (a) $x \notin C$, since $M_i = \emptyset$, we have $M_i^s = \emptyset$. From condition (ii), $B_v(x) \neq \emptyset$. From the above construction procedure, $\exists\, x_1 \in X_{i-1}$ such that $x_1 \in B_v(x)$. Case (b) $x \in C$, since $M_i = \emptyset$, we have $M_i^c = \emptyset$. From the above construction procedure, $\exists\, x^* \in X_{i-1}$ such that $x^* \in B(x)$. By Assumption 3.1, we have $x^* \in B_v(x)$. By cases (a) and (b), we get that for each $x \in X_{i-1}$, there is a $y \in X_{i-1}$, such that $y \in B_v(x)$. Because of the finiteness of X_{i-1}, we may get a sequence satisfying Eq. (1), which contradicts condition (iv).

Now, consider X_1. If $X_1 = X_0 \backslash M_1 = \emptyset$, then $X_i = \emptyset, M_{i+1} = \emptyset, 1 \leqslant i \leqslant n-1$. If $X_1 \neq \emptyset$, then $M_2 \neq \emptyset$. Repeating the above consideration on X_2 and so on, we can get a sequence $(M_1, M_2, \ldots, M_r)$ such that $M_i \neq \emptyset$, $1 \leqslant i \leqslant r$, and $M_j = \emptyset,\ r+1 \leqslant j \leqslant n, 1 \leqslant r \leqslant n$. Let

$$M_1 = (x_1^1, \ldots, x_{n1}^1), \ldots, M_r = (x_1^r, \ldots, x_{nr}^r)$$

in which we arrange the nodes in such a way that $\forall x \in C \cap M_i, y \in B(x) \cap M_i$ is before x in $M_i (i = 1, 2, \ldots, r)$. Then from the above construction procedure, we know that $(M_1, M_2, \ldots, M_r) = (x_1^1, \ldots, x_{n1}^1; \ldots; x_1^r, \ldots, x_{nr}^r)$ with respect to V is an SEP. □

Remark 3.2. The constructive proof for sufficiency above is different from that of Theorem 3.1 in [6]. The proof here is a *forward* method, not a backward method of [6]. The proof is much easier to understand than that of [6]. Above all, the SEP gives us a clear ordering of expansion process; we will further discuss this later.

Remark 3.3. The assumption (iv) in Theorem 3.1 is that to avoid cycling in an SEP, we will use the system of inequalities similar to the method of Miller et al. [9] as its equivalent statement (assumption (iv) in the forthcoming Theorem 3.2, or assumptions (iv) and (iv)* in the forthcoming Theorem 3.3).

Definition 3.2. Let X, $X \subset \text{HD}\backslash\text{Sk}$, be a set of skills, $V\,(V \subset U)$ a set of arcs, $X^* = X \cup \{x \in \text{Sk} | A_v(x) \cap X \neq \emptyset\}$, define

$$w_i = \begin{cases} 1 & \text{if } x_i \in X^*, \\ 0 & \text{otherwise.} \end{cases}$$

$$y(i,j) = \begin{cases} 1 & \text{if } (x_i, x_j) \in V, \\ 0 & \text{otherwise.} \end{cases}$$

Theorem 3.2. *Let X be a set of skills* ($X \subset$ HD\Sk), *and V, a set of arcs* ($V \subset U$). *Then X and V form an SEP if and only if* w_i *and y(i, j)* (*as in Definition 3.2*) *satisfy*:

(i) $w_i = 1,\ \forall x_i \in \mathrm{Tr}\backslash\mathrm{Sk}$,
(ii) $\sum_{x_i \in B(x_j)} y(i,j) = w_j,\ \forall x_j \in \mathrm{HD}\backslash(C \cup \mathrm{Sk})$,
(ii)* $[\underline{A}(x_i)]w_i \geqslant \sum_{x_j \in \underline{A}(x_i)} y(i,j),\ \forall x_i \in \mathrm{HD}$,
(iii) $[B(x_c)]w_c \leqslant \sum_{x_{ci} \in B(x_c)} w_{ci}, \forall x_c \in C\backslash\mathrm{Sk}$
(*where* $[A(x_i)]$ *and* $[B(x_c)]$ *are the numbers of nodes in* $A(x_i)$ *and* $B(x_c)$, *respectively*),
(iv) *there exists a* $u_i = u(x_i)$ *for each* $x_i \in$ HD\Sk, *and* $u_i \in \{0, 1, 2, \ldots, n\}$ *such that* $u_i - u_j + (n+1)y(i,j) \leqslant n,\ x_i, x_j \in \mathrm{HD}\backslash\mathrm{Sk}$, *n is the number of nodes in* HD\Sk.

Proof. We will prove that conditions (i)–(iv) of Theorem 3.1 are equivalent to the conditions (i)–(iv) of Theorem 3.2. From Definition 3.2, it is easy to check that condition (i) of Theorem 3.2 is equivalent to Tr\Sk $\subset X$ which is the condition (i) of Theorem 3.1.

Now, let us prove the equivalence of conditions (ii)–(ii)* of Theorem 3.2 and condition (ii) of Theorem 3.1. For each $x_j \in X\backslash C$, from Definition 3.2, $w_j = 1$. By condition (ii) of Theorem 3.2, we have that there is only one skill $x_i \in B(x_j)$ such that $y(i, j) = 1$. That is, $(x_i, x_j) \in V$, or $\{x_i\} = B_v(x_j)$. By (ii)* of Theorem 3.2, we have $w_i = 1$, i.e., $x_i \in X^*$. Thus condition (ii) of Theorem 3.1 holds.

For each $x_j \in$ HD\($C \cup$ Sk), considering two possible cases: Case (a) $x_j \in X$, i.e., $w_j = 1$, by (ii) of Theorem 3.1, we know $B_v(x_j)$ is a singleton. If we let $\{x_i\} = B_v(x_j)$, then $(x_i, x_j) \in V$ is the unique arc such that $y(i, j) = 1$. Thus condition (ii) of Theorem 3.2 holds. Case (b) $x_j \notin X$, i.e., $w_j = 0$, for every $x_i \in B_v(x_j)$, by Assumption 3.1, we have $(x_i, x_j) \notin V$, or $y(i, j) = 0$. Thus condition (ii) of Theorem 3.2 holds also. Similarly, it is easy to check that condition (ii)* of Theorem 3.2 holds.

Next, let us prove the equivalence of condition (iii) of Theorem 3.2 and condition (iii) of Theorem 3.1. For each $x_c \in X \cap C$, from Definition 3.2, $w_c = 1$. By (iii) of Theorem 3.2, we have $\forall x_{ci} \in B(x_c)$, $w_{ci} = 1$. That is, $B(x_c) \subset X^*$. Thus condition (iii) of Theorem 3.1 holds.

For each $x_c \in C$ \Sk, consider two possible cases: Case (a) $x_c \in X$, i.e., $w_c = 1$, by condition (iii) of Theorem 3.1, we have $B(x_c) \subset X^*$. That is, $\forall x_{ci} \in B(x_c), w_{ci} = 1$. Thus condition (iii) of Theorem 3.2 holds. Case (b) $x_c \notin X$, by $X \subset$ HD\Sk, we have $x_c \notin X^*$, or $w_c = 0$. In this case, condition (iii) of Theorem 3.2 holds clearly.

Finally, we will prove the equivalence of condition (iv) of Theorem 3.2 and condition (iv) of Theorem 3.1 under the assumption that conditions (i)–(iii) of Theorem 3.1 or conditions (i)–(iii) of Theorem 3.2 hold.

If there is a sequence $(x_{p1}, x_{p2}, \ldots, x_{pk}) \subset X$ satisfying

$$x_{p1} \in B_v(x_{p2}), \ldots, x_{pk-1} \in B_v(x_{pk}), x_{pk} \in B_v(x_{p1}).$$

Then

$$y(p_1, p_2) = y(p_2, p_3) = \cdots = y(p_{k-1}, p_k) = y(p_k, p_1) = 1.$$

From (iv) of Theorem 3.2, we have

$$u_{p1} - u_{p2} + n + 1 \leqslant n,$$
$$u_{p2} - u_{p3} + n + 1 \leqslant n,$$
$$\ldots\ldots\ldots\ldots\ldots\ldots$$
$$u_{pk} - u_{p1} + n + 1 \leqslant n.$$

Summing up the above inequalities, we obtain $n + 1 \leqslant n$, which is a contradiction. Thus the condition (iv) of Theorem 3.1 holds.

If conditions (i)–(iv) of Theorem 3.1 hold, then X and V form an SEP with a sequence ψ defined as follows:

$$\psi = (M_1, M_2, \ldots, M_r) = (x_1^1, \ldots, x_{n1}^1; \ldots; x_1^r, \ldots, x_{nr}^r).$$

Let $u_i^j = u(x_i^j)(i = 1, 2, \ldots, n_j, j = 1, 2, \ldots, r)$ be the ordering number of the position of x_i^j in ψ; i.e., $(u_1^1, \ldots, u_{n1}^1; \ldots; u_1^r, \ldots, u_{nr}^r) = (1, \ldots, n_1; n_1 + 1, \ldots, n_1 + n_2; \ldots; n_1 + \cdots + n_{r-1} + 1, \ldots, n_1 + \cdots + n_r)$.

For x_i, $x_j \in$ HD\Sk, consider two possible cases: Case (a) x_i, $x_j \in X$, i.e., x_i, $x_j \in \{\psi\}$. Then either $y(i, j) = 1$ or $y(i, j) = 0$. From the above definition of u_i^j, condition (iv) of Theorem 3.2 holds. Case (b) x_i or $x_j \notin X$, i.e., $y(i, j) = 0$. Then condition (iv) of Theorem 3.2 holds too. Summing up all of the above, we complete the proof. □

Remark 3.4. Conditions (ii) and (ii)* of Theorem 3.2 show close relationships among w_i, w_j and $y(i, j)$. That is, if $y(i, j) = 1$, or $(x_i, x_j) \in V$, then from (ii), we have $w_j = 1$, or $x_j \in X^*$; and from condition (ii)*, we have $w_i = 1$, or $x_i \in X^*$. Similar to the proof of Lemma 3.1 of [5], $\forall x_j \in X \backslash (C \cup \text{Sk})$, we can find a path from some nodes (which may have more than one since there are compound nodes in HD) in Sk to x_j in SEP. Indeed, if $w_j = 1$, then condition (ii) implies: $\exists\, x_i \in B(x_j)$ such that $y(i, j) = 1$. Then condition (ii)* assumes that $w_i = 1$, or $x_i \in X^*$. If $x_i \in$ Sk, then we get the path. Otherwise, continue the same procedure. Because of the finiteness of HD, we either get a sequence of nodes satisfying

$$x_{p1} \in B_v(x_{p2}), \ldots, x_{pk-1} \in B_v(x_{pk}), x_{pk} \in B_v(x_{p1}),$$

which is impossible because of condition (iv) (or condition (iv) of Theorem 3.1); or get a path from Sk to x_j. Conditions (ii) and (ii)* seem to be simpler than that of (iii) and (iv) of Theorem 3.1 in [5].

To avoid cycling, we introduce another system of inequalities and get the following theorem.

Theorem 3.3. *Let X be a set of skills* ($X \subset$ HD\Sk), *and V, a set of arcs* ($V \subset U$). *Then X and V form an SEP if and only if* w_i *and* $y(i, j)$ (*as in Definition 3.2*) *satisfy*:

(i) $w_i = 1, \quad \forall x_i \in \text{Tr}\backslash\text{Sk}$,

(ii) $\sum_{x_i \in B(x_j)} y(i, j) = w_j, \quad \forall x_j \in \text{HD}\backslash(C \cup \text{Sk})$,

(ii)* $[\underline{A}(x_i)]w_i \geqslant \sum_{x_j \in \underline{A}(x_i)} y(i, j), \quad \forall x_i \in \text{HD}$,

(iii) $[B(x_c)]w_c \leq \sum_{x_{ci} \in B(x_c)} w_{ci}, \quad \forall x_c \in C\backslash\text{Sk}$,

(*where* $[A(x_i)]$ *and* $[B(x_c)]$ *are the numbers of nodes in* $A(x_i)$ *and* $B(x_c)$, *respectively*),

(iv) *for each* $x_i \in \text{HD}\backslash\text{Sk}$, *there exists an integer* $u_i = u(x_i)$ *such that the following holds*: $u_i - u_j + ny(i, j) \leqslant n - 1, \quad w_i \leqslant u_i \leqslant nw_i$, where $x_i \in \text{HD}\backslash\text{Sk}$, $x_j \in \text{HD}\backslash(C \cup \text{Sk})$, *and n is the number of nodes in* HD\\$(C \cup$ Sk),

(iv)* *for every* $x_c = x_{c1} \wedge x_{c2} \wedge \cdots \wedge x_{ck} \in C\backslash\text{Sk}$, *if* $x_{ci} \notin$ Sk, *then* $u_c + n(k - \sum_{j=1}^{k} w_{cj}) \geqslant u_{ci}$ (*k is the number of singleton skills compounded for* x_c).

Proof. Comparing Theorems 3.3 with 3.2, we need only to prove the equivalence of conditions (iv) and (iv)* of Theorem 3.3 and condition (iv) of Theorem 3.1 under the assumption that conditions (i)–(iii) of Theorem 3.1 or conditions (i)–(iii) of Theorem 3.3 hold.

Suppose there is a sequence $(x_{p1}, x_{p2}, \ldots, x_{pk}) \subset X$ satisfying

$$x_{p1} \in B_v(x_{p2}), \ldots, x_{pk-1} \in B_v(x_{pk}), x_{pk} \in B_v(x_{p1}).$$

Consider two possible cases: Case (a) $x_{pi} \notin C, (x_{pi-1}, x_{pi}) \in V$(i.e., $y(p_{i-1}, p_i) = 1$). By (iv) of Theorem 3.3, we have $u_{pi} > u_{pi-1}$. Case (b) $x_{pi} \in C, (x_{pi-1}, x_{pi}) \in V$(i.e., $y(p_{i-1}, p_i) = 1$). By Definition 2.1, we have $x_{pi-1} \notin C$, from case (a), we have $u_{pi-1} > u_{pi-2}$. By (iv)* of Theorem 3.3, we have $u_{pi} \geqslant u_{pi-1}$. Considering all x_{pi} in $(x_{p1}, x_{p2}, \ldots, x_{pk})$, we can get either

$$u_{pi} > u_{pi-1} \geqslant u_{pi-2} \geqslant \cdots \geqslant u_{pi}, \quad \text{or} \quad u_{pi} \geqslant u_{pi-1} > u_{pi-2} \geqslant \cdots \geqslant u_{pi}.$$

Each of these leads to a contradiction. Therefore, condition (iv) of Theorem 3.1 holds.

If conditions (i)–(iv) of Theorem 3.1 hold, then X and V form an SEP with a sequence ψ defined by

$$\psi = (M_1, M_2, \ldots, M_r) = (x_1^1, \ldots, x_{n1}^1; \ldots; x_1^r, \ldots, x_{nr}^r).$$

Let $u_i^j = u(x_i^j)(i = 1, 2, \ldots, n_j, j = 1, 2, \ldots, r)$ be the ordering number of the position of x_i^j in ψ. Define the sequence $(\{u_1^1, \ldots, u_{n1}^1; \ldots; u_1^r, \ldots, u_{nr}^r\} \backslash \{u_i^j = u(x_i^j) | x_i^j \in C\}) = (1, 2, \ldots, m)$, where m is the number of singleton nodes in ψ. We see that by setting

$$u_c = \max\{u_{ci} = u(x_{ci}) | x_{ci} \in B(x_c) \backslash \mathrm{Sk}\}, \text{ if } x_c \in X \cap C; \quad \text{and}$$

$$u_i = u(x_i) = 0, \text{ if } x_i \notin X.$$

Then conditions (iv) and (iv)* are satisfied. □

Remark 3.5. In addition to condition (iv), condition (iv)* in Theorem 3.3 is also necessary to avoid cycles. Let us consider the figure (Fig. 3), where $x_5 = x_3 \wedge x_4, x_i \notin \mathrm{Sk}, i = 1, 2, 3, 4, 5$. Let $u_1 = u(x_1) = 2, u_2 = u(x_2) = 2, u_3 = u(x_3) = 3, u_4 = u(x_4) = 3, u_5 = u(x_5) = 1$. If Fig. 3 represents a part of the expansion process, it is easy to check that condition (iv) in Theorem 3.3 is satisfied, but condition (iv)* is not satisfied. Obviously, the digraph in Fig. 3 has a cycle.

4. Minimal cost stage expansion process

In this section, we first introduce the concept of the minimal cost SEP; then we will prove that the total cost of a minimal cost SEP is equal to that of the minimal cost expansion process (see Definition 2.2). Therefore, the introduction of SEP concept is useful. We then provide a method to find a minimal cost SEP by using integer programming. Finally, an example is provided to illustrate the method.

Remark 4.1. Let $\psi = (x_{k1}, x_{k2}, \ldots, x_{kn})$ be an expansion process with

$$C_j(\psi) = \min\{c(x, x_{kj}) | x \in Sk_{j-1}(\psi)\} = c(x_{ki}, x_{kj}), \quad i < j.$$

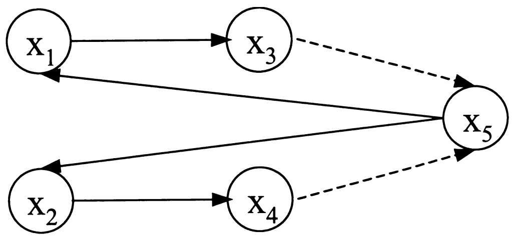

Fig. 3. A cyclic connection.

Define

$$V_\psi = \{(x_{ki}, x_{kj}) | c(x_{ki}, x_{kj}) = C_j(\psi), x_{kj} \notin C, 1 \leqslant j \leqslant n\} \cup \{(x_{ki}, x_{kj}) | x_{ki} \in B(x_{kj}), x_{kj} \in C, 1 \leqslant j \leqslant n\}.$$

Rearranging the ψ with V_ψ according to the construction procedure in the proof for sufficiency of Theorem 3.1, we get that $\psi^* = (x_1^1, \ldots, x_{n1}^1; \ldots; x_1^r, \ldots, x_{nr}^r)$ with respect to V_ψ is an SEP.

Observe that

$$C(\psi) = C_1(\psi) + \cdots + C_n(\psi) = \sum_{(x_{ki}, x_{kj}) \in V_\psi} c(x_{ki}, x_{kj}). \tag{2}$$

Definition 4.1. Let $\psi = (M_1, M_2, ..., M_r) = (x_1^1, \ldots, x_{n1}^1; \ldots; x_1^r, \ldots, x_{nr}^r)$ with respect to V be an SEP. We call $C_{j,i}^s(\psi, V) = c(B_v(x_i^j), x_i^j)(1 \leqslant i \leqslant n_j, 1 \leqslant j \leqslant r)$ the ith step expansion cost in jth stage M_j of the SEP (here $x_i^j \notin C$, if $x_i^j \in C$, then $C_{j,i}^S(\psi, V) = 0$);

$$C_j^s(\psi, V) = \sum_{i=1}^{n_j} C_{j,i}^S(\psi, V) \text{ the } j\text{th stage-expansion cost of the SEP;}$$

$$C^S(\psi, V) = \sum_{j=1}^{r} C_j^S(\psi, V) = \sum_{(x,y) \in V} c(x, y) \text{ the total cost of the SEP.}$$

(When there is no confusion, we simplify the notations $C_{j,i}^s(\psi, V), C_j^s(\psi, V)$, and $C^s(\psi, V)$ into $C_{j,i}^s(\psi), C_j^s(\psi)$, and $C^S(\psi)$.)

We call ψ^* with respect to V^* a minimal cost SEP, if $C^S(\psi^*) = \min\{C^S(\psi) | \psi \text{ with respect to } V \text{ is an SEP}\}$.

Remark 4.2. Let ψ with respect to V be an SEP. In general

$$C_{j,i}^S(\psi) \neq \min\{c(x, x_i^j) | x \in \text{Sk} \cup M_1 \cup \cdots \cup M_{j-1} \cup \{x_1^j, \ldots, x_{ni-1}^j\}\}.$$

For a counter example, please refer to Remark 4.1 of [6]. Now, let us give the following result.

Proposition 4.1. *Let $\psi = (M_1, M_2, \ldots, M_r) = (x_1^1, \ldots, x_{n1}^1; \ldots; x_1^r, \ldots, x_{nr}^r)$ with respect to V be a minimal cost SEP. Then*

$$C_{j,i}^s(\psi) = \min\{c(x, x_i^j) | x \in Sk \cup M_1 \cup \cdots \cup M_{j-1} \cup \{x_1^j, \ldots, x_{ni-1}^j\}\}, \quad \text{for} \quad 1 \leqslant i \leqslant n_j, 1 \leqslant j \leqslant r.$$

Proof. If $x_i^j \in C$, then $C_{j,i}^s(\psi) = 0$ because $c(B(x_i^j), x_i^j) = 0$ (see Remark 2.1). Suppose there is a $x_i^j \notin C$, such that

$$C_{j,i}^s(\psi) \neq \min\{c(x, x_i^j) | x \in Sk \cup M_1 \cup \cdots \cup M_{j-1} \cup \{x_1^j, \ldots, x_{ni-1}^j\}\}.$$

Let $C_{j,i}^s(\psi) = c(x_k^{j-1}, x_i^j)$. Then $\exists\, z \in \text{Sk} \cup \{x_1^1, \ldots, x_{n1}^1; \ldots; x_1^j, \ldots, x_{i-1}^j\}, z \neq x_k^{j-1}$ such that $c(z, x_i^j) < c(x_k^{j-1}, x_i^j)$.

Let $V^* = (V\{(x_k^{j-1}, x_i^j)\}) \cup \{(z, x_i^j)\}$. Then, by rearranging ψ with V^* according to the construction procedure in the proof for sufficiency of Theorem 3.1, we get a new sequence ψ^* such that ψ^* with respect to V^* is also an SEP. This implies that

$$C^S(\psi^*) = \sum_{(x,y) \in V} c(x, y) - c(x_k^{j-1}, x_i^j) + c(z, x_i^j) < \sum_{(x,y) \in V} c(x, y) = C^S(\psi)$$

which contradicts the assumption. $\square$

Combining Remark 4.1, Eq. (2) and Definition 4.1 with Proposition 4.1, we get Theorem 4.1.

Theorem 4.1. min $C^S(\psi) = \min\ C(\psi)$.

From Theorem 3.2, Theorem 3.3 and Definition 4.1, we get the following theorem by which we can find a minimal SEP.

Theorem 4.2. *Let X be a set of skills* ($X \subset$ HD\Sk), *and V, a set of arcs* ($V \subset U$). *Then X and V form a minimal cost SEP if and only if the corresponding* $\{w_i\}$ *and* $\{y(i, j)\}$ (*as in Definition* 3.2) *solve the following integer programming problem*:

$$\min z = \sum c(x_i, x_j) y(i,j), \quad \text{subject to}$$

Conditions(i)–(iv)of Theorem 3.2;

or

Conditions(i)–(vi)*of Theorem 3.3.

Remark 4.3. From Theorem 4.2, we can find a minimal SEP; but we cannot acquire the ordering number of stage simultaneously. Following from Theorem 3.3, we will give an integer program to find a minimal SEP and at the same time get the ordering number of the stage for each node.

Theorem 4.3. *Let X be a set of skills* ($X \subset$ HD\Sk), *and V, a set of arcs* ($V \subset U$). *Then X and V form a minimal cost SEP if and only if the corresponding* $\{w_i\}$, $\{y(i, j)\}$ (*as in Definition* 3.2), *together with some integers* $\{u_i\}$ *solve the following integer programming Problem A*.

Problem A. min $z = \sum c(x_i, x_j) y(i,j) + \varepsilon\mu$, where ε ($\varepsilon > 0$) is a sufficiently small constant,
subject to:

(i) $w_i = 1, \quad \forall x_i \in \text{Tr}\backslash\text{Sk}$,

(ii) $\sum_{x_i \in B(x_j)} y(i,j) = w_j, \quad \forall x_j \in \text{HD}\backslash(C \cup \text{Sk})$,

(ii)* $[A(x_i)] w_i \geqslant \sum_{x_j \in \underline{A}(x_i)} y(i,j), \quad \forall x_i \in \text{HD}$,

(iii) $[B(x_c)] w_c \leqslant \sum_{x_{ci} \in B(x_c)} w_{ci}, \quad \forall x_c \in C\backslash\text{Sk}$,
(where $[A(x_i)]$ and $[B(x_c)]$ are the numbers of nodes in $A(x_i)$ and $B(x_c)$, respectively),

(iv) for each $x_i \in$ HD\Sk, there exists an integer $u_i = u(x_i)$ such that the following holds: $u_i - u_j + ny(i,j) \leqslant n - 1, \quad w_i \leqslant u_i \leqslant nw_i$,
where $x_i \in$ HD\Sk, $x_j \in$ HD\($C \cup$ Sk), and n is the number of nodes in HD\($C \cup$ Sk);

(iv)* for every $x_c = x_{c1} \wedge x_{c2} \wedge \ldots \wedge x_{ck} \in C\backslash\text{Sk}$, *if* $x_{ci} \notin$ Sk, then $u_c + n(k - \sum_{j=1}^{k} w_{cj}) \geqslant u_{ci}$ (k is the number of singleton skills compounded for x_c);

(v) $\mu = \sum_{x_i \notin Sk} u_i$,
where $\{w_i\}$ and $\{y(i, j)\}$ are $0-1$ variables, while $\{u_i\}$ and μ are integer variables.

The solution u_i represents the stage ordering number of x_i in the minimal cost SEP. If $z_0 = \min z$, and the corresponding $\mu = \mu_0$, then $z_0 - \varepsilon\mu_0$ is the total cost of the SEP.

Proof. By comparing Theorem 4.2 with Theorem 3.3, it suffices to show that we can get a minimal cost SEP and at the same time determine the ordering number of the stage of each node by solving Problem A.

Let $t = \sum c(x_i, x_j) y(i,j), z_0 = t_0 + \varepsilon\mu_0 = \min z$. Suppose there is a solution of Problem A with $z_1 = t_1 + \varepsilon\mu_1$ such that $t_1 < t_0$. Consider

$$z_1 - z_0 = t_1 - t_0 + \varepsilon(\mu_1 - \mu_0).$$

From (iv)–(v), we know that $\mu_1 - \mu_0$ is a finite number. By choosing ε to be a sufficiently small positive number such that $z_1 - z_0 < 0$, we will get a contradiction. Therefore if we minimize z, we will get a minimal cost SEP.

From (iv), $w_i \leqslant u_i \leqslant n w_i$, we have if $x_i \in (\mathrm{HD}\backslash\mathrm{Sk})\backslash X$, then $w_i = 0$. Therefore, $u_i = 0$. From $u_i - u_j + (n+1) y(i,j) \leqslant n$, we have if $x_i \in \mathrm{HD}\backslash\mathrm{Sk}$, $x_j \in \mathrm{HD}\backslash(C \cup \mathrm{Sk})$, $(x_i, x_j) \in V$ (i.e., $y(i,j) = 1$), then $u_i - u_j + n + 1 \leqslant n$. That is, $u_i + 1 \leqslant u_j$. From (iv)*, we have if $x_c = x_{c1} \wedge x_{c2} \wedge \ldots \wedge x_{ck} \in C \cap X \backslash \mathrm{Sk}$ and $x_{ci} \notin \mathrm{Sk}$, then $u_c \geqslant u_{ci}$. Note that by minimizing z, we also minimize μ. If $\{w_i\}$, $\{y(i,j)\}$ and $\{u_i\}$ is a solution to Problem A, then from (v), $u_c = \max\{u_{ci} | x_{ci} \notin \mathrm{Sk}\}$ (refer to Remark 3.1). Thus u_i represents the ordering number of the stage of x_i. Note that if we get $z = z_0$ and $\mu = \mu_0$ by solving Problem A, then $z_0 - \varepsilon\mu_0$is the total cost of the minimal cost SEP. □

Remark 4.4. In Theorem 4.3, we let ε be small enough so that in minimizing $\sum c(x_i, x_j) y(i,j) + \varepsilon\mu$, we will minimize $\sum c(x_i, x_j) y(i,j)$ first (to get a minimal cost SEP) and then minimize μ to get the solution u_i which represents the stage ordering number of x_i in the minimal cost SEP. Note: in previous papers [4,6], u_i does not have such a meaning as the stage ordering number of x_i. In addition, "ε is sufficiently small" is relative to each cost function $c(x_i, x_j)$. Notice that $\mu = \sum_{x_i \notin Sk} u_i$ is independent of the cost functions between the skills. If each cost function is relatively large, the problem can be solved by setting ε to be a small integer. If there is a small cost function, then we can solve the problem by first multiplying each cost function by a bigger positive integer, such as 10, or 100 etc., and choosing ε to be a small integer.

Example 4.1. Now we will use the method in Theorem 4.3 to solve Example 2.1.

For convenience, let $x_1 = a$, $x_2 = b$, $x_3 = c$, $x_4 = a \wedge b$, $x_5 = d$, $x_6 = e$, $x_7 = f$, $x_8 = e \wedge f$, $x_9 = g$, $x_{10} = h$; $y_1 = y(1,5)$, $y_2 = y(1,7)$, $y_3 = y(1,9)$, $y_4 = y(2,5), \ldots, y_{20} = y(8,9)$, $y_{21} = y(8,10)$. $u_i = u_i(x_i)$, $i = 5, 6, \ldots, 10$.

$$\begin{aligned} \min z \ &= y_1 + 3y_2 + 4y_3 + 2y_4 + 4y_5 + 2y_6 + 3y_7 + y_8 + 2.5y_9 + 3.5y_{10} + y_{11} + 1.8y_{12} + 4y_{13} + 4y_{14} + 2y_{15} \\ &\quad + 2y_{16} + 2y_{17} + 2y_{18} + 2y_{19} + 1.5y_{20} + 1.5y_{21} + \varepsilon\mu \end{aligned}$$

subject to

$$\begin{aligned}
& w_5 = w_7 = w_9 = w_{10} = 1, \\
& \quad y_1 + y_4 + y_6 + y_8 + y_{15} + y_{19} = w_5, \\
& \quad y_7 + y_{11} + y_{16} = w_6, \\
& \quad y_2 + y_9 + y_{12} = w_7, \\
& \quad y_3 + y_5 + y_{10} + y_{13} + y_{17} + y_{20} = w_9, \\
& \quad y_{14} + y_{18} + y_{21} = w_{10}, \\
& \quad 3w_1 \geqslant y_1 + y_2 + y_3, \\
& \quad 2w_2 \geqslant y_4 + y_5, \\
& \quad 2w_3 \geqslant y_6 + y_7, \\
& \quad 3w_4 \geqslant y_8 + y_9 + y_{10}, \\
& \quad 2w_5 \geqslant y_{11} + y_{12}, \\
& \quad 2w_6 \geqslant y_{13} + y_{14}, \\
& \quad 4w_7 \geqslant y_{15} + y_{16} + y_{17} + y_{18}, \\
& \quad 3w_8 \geqslant y_{19} + y_{20} + y_{21},
\end{aligned}$$

$$2w_8 \leqslant w_6 + w_7,$$
$$u_5 - u_6 + 5y_{11} \leqslant 4,$$
$$u_5 - u_7 + 5y_{12} \leqslant 4,$$
$$u_6 - u_9 + 5y_{13} \leqslant 4,$$
$$u_6 - u_{10} + 5y_{14} \leqslant 4,$$
$$u_7 - u_5 + 5y_{15} \leqslant 4,$$
$$u_7 - u_6 + 5y_{16} \leqslant 4,$$
$$u_7 - u_9 + 5y_{17} \leqslant 4,$$
$$u_7 - u_{10} + 5y_{18} \leqslant 4,$$
$$u_8 - u_5 + 5y_{19} \leqslant 4,$$
$$u_8 - u_9 + 5y_{20} \leqslant 4,$$
$$u_8 - u_{10} + 5y_{21} \leqslant 4,$$
$$w_i \leqslant u_i \leqslant 5w_i, \quad i = 5 \sim 10;$$
$$u_8 + 5(2 - w_6 - w_7) \geqslant u_6,$$
$$u_8 + 5(2 - w_6 - w_7) \geqslant u_7,$$
$$\mu = u_5 + u_6 + u_7 + u_8 + u_9 + u_{10}.$$

Here $y_i (i = 1 \sim 21)$ and $w_i (i = 1 \sim 10)$ are 0−1 variables, $u_i (i = 5 \sim 10)$ are integer variables, and ε is a sufficiently small constant. In this example, we set $\varepsilon = 0.1$.

Using integer programming package [10], we obtain the solution as

$$w_4 = w_5 = w_7 = w_9 = w_{10} = 1;\ y_8 = y_{12} = y_{17} = y_{18} = 1;$$
$$u_5 = 1,\ u_7 = 2,\ u_9 = 3, u_{10} = 3;\ \mu = 9,\ z = 7.7.$$

$z - 0.1\mu = 7.7 - 0.9 = 6.8$ is the minimal total cost. Fig. 4 depicts the minimal cost SEP.

Suppose we change the cost from d to e. For example, let $c(d, e) = 0.5$, while the other conditions remain the same. This is equivalent to changing the coefficient of y_{11} in the objective function from 1 to 0.5. By solving the problem, we get

$$w_1 = w_5 = w_6 = w_7 = w_8 = w_9 = w_{10} = 1;\ y_1 = y_{11} = y_{12} = y_{20} = y_{21} = 1;$$
$$u_5 = 1,\ u_6 = u_7 = u_8 = 2,\ u_9 = u_{10} = 3;\ \mu = 13,\ z = 7.6.$$

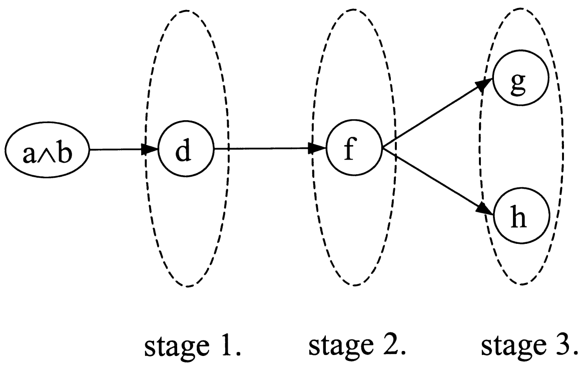

Fig. 4. The first minimal cost SEP.

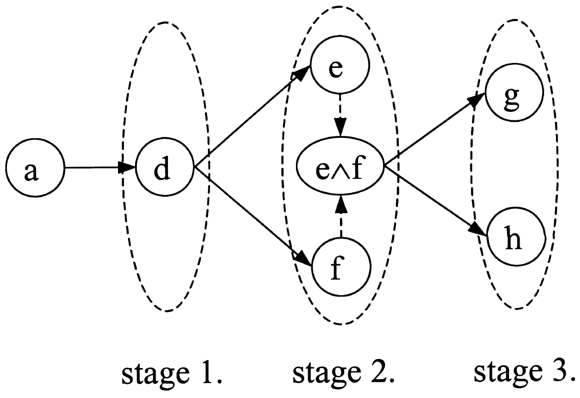

Fig. 5. The second minimal cost SEP.

$z - 0.1\mu = 7.6 - 1.3 = 6.3$ is the minimal total cost. The corresponding minimal cost SEP is depicted in Fig. 5.

5. Conclusion

We have discussed a new method of finding the optimal expansion process by introducing the concept of the SEP. This method makes it possible to solve those competence set expansion problems which have compound skills, intermediate skills and cyclic connections among skills. This method can determine the stage ordering of each skill as well as find an optimal expansion process. The SEP gives us a clear ordering of the expansion process. The SEP method can be used to design optimal competence sets, similar to those of [6]; we leave it to the readers to explore.

References

[1] P.L. Yu, Forming Winning Strategies: An Integrated Theory of Habitual Domains, Springer, Berlin, 1990.
[2] P.L. Yu, Habitual Domains: Freeing Yourself from the Limits on Your Life, Highwater Editions, Kansas City, MO, 1995.
[3] P.L. Yu, D. Zhang, A foundation for competence set analysis, Mathematical Social Sciences 20 (1990) 251–299.
[4] H.L. Li, Incorporating competence sets of decision makers by deduction graph, Forthcoming in Operations Research.
[5] H.L. Li, P.L. Yu, Optimal competence set expansion using deduction graphs, Journal of Optimization Theory and Applications 80 (1994) 75–91.
[6] D.S. Shi, P.L. Yu, Optimal expansion and design of competence sets with asymmetric acquiring costs, Journal of Optimization Theory and Applications 88 (1996) 643–658.
[7] P.L. Yu, D. Zhang, Optimal expansion of competence set and decision support, Information System and Operational Research 30 (1992) 68–84.
[8] P.L. Yu, D. Zhang, A marginal analysis for competence set expansion, Journal of Optimization Theory and Applications 76 (1993) 87–109.
[9] C.E. Miller, A.W. Tucker, R.A. Zemlin, Integer programming formulation of traveling salesman problems, Journal of the Association for Computing Machinery 7 (1960) 326–329.
[10] L. Schrage, Linear, Integer, and Quadratic Programming with LINDO, Scientific Press, Palo Alto, CA, 1984.