Research on Hybrid Assessment Algorithm based on Local Search for Risk Management in Virtual Enterprise

Deng Wei
School of Science & Arts, Shanghai Maritime University, Shanghai, P. R. China
Email: yunfuhuanghe@126.com

Neng Wang
School of Economics and Management, Dezhou University, Dezhou, P. R. China
Email: nengwang234@163.com

Abstract—**This paper presents a hybrid assessment algorithm based on local search for the risk management problem in virtual enterprise. Using this method, the entire risk level of the virtual enterprise is minimized by combination optimization of risk solutions with the constraint of risk investment. Results of numerical tests show the effectiveness of the algorithm and it is the scientific management method for virtual enterprise risk management.**

Index Terms—**Virtual Enterprise, Risk Management, Local Search**

I. Introduction

With the emergence of a global economy and trend towards customer customization, manufactures have been seeking new paradigms, such as lean production, agile manufacturing, and virtual enterprises (VEs), to grasp market opportunities in a competitive global environment. A virtual enterprise can be considered as a temporary alliance of globally distributed independent enterprises that participates in the different phases of the life cycle of a product or service, and work to share resources, skills, and costs, supported by Information and Communication Technologies (ICT),in order to better take advantage of market opportunities and successfully carry out a responsible corporate strategy(Bernus and Nemes, 1999).It can be defined as ‘‘a subset of units and processes within the supply chain network, consisting of a matrix of largely co-operating manufacturing, stores, and transport units of mixed ownership, which behave like a single company through strong co-ordination and co-operation towards mutual goals’’(Makatsoris et al.,1996). On the one hand, VEs can help enterprises to respond rapidly to market demand by sharing capabilities, resources, and so forth. On the other hand, enterprises in a VE face more risks than a stand-alone enterprise. Many factors, such as delivery performance, price and demand, etc., can cause risks. The risks under a network manufacturing environment have been classified by a number of authors in order to better analyze them. Treleven and Schweikhart(1988)have classified the risks into five categories connected with disruption, price, inventories and schedule, technology, and quality. Other risks, mentioned by Virolainen and Tuominen(1998)are associated with availability, configurations and currencies. Hallikas et al.(2004)grouped the risks in four types: too low or inappropriate demand, problems in fulfilling customer deliveries, cost management and pricing, and weaknesses in resources. Hence, risk management is the key problem to overcome in a VE in order to ensure success. Risk programming, an important stage of risk management, is the process used to determine the risk management strategy and to realize concrete measures and means. In this process, the known risk is eliminated as soon as possible (Fan et al., 2008). As a VE is a complex system temporarily composed of many stand-alone enterprises due to market opportunities, the traditional risk model no longer works for a VE(Park and Favrel,1999;Hallikas et al.,2004).

Hence, much attention is being paid to risk manage ment in a VE. Hallikas et al.(2004)have proposed risk management processes in supplier networks. Wang and Tang(2002)established an optimization model for a VE to reduce its risk and increase its income. Ip et al.(2003) studied the optimization model for minimizing risk in partner selection while ensuring the due date of a project. In these studies, the characteristics of the project organization mode and the stochastic features of events in VE are considered. However, another main characteristic of a VE is that, with regard to risk management, there are always no historical materials to refer to and reliance has mainly been placed on the experiences and subjective judgments of the personnel, which are fuzzy. In view of the fact that the model of risk programming for a VE is nonlinear and discrete, we have developed a tabu search algorithm in which a fuzzy hybrid assessment is embedded. Experiences with computation have demonstrated that it is an efficient method of finding the optimal solution.

Corresponding author at: School of Economics and Management, Dezhou University, Dezhou, Shandong Province, China, P.C. 253023

doi:10.4304/jsw.5.12.1409-1416

In Section 2, the risk programming for a VE is proposed, with a focus on the project organization mode and the fuzzy characteristics of a VE, which are different from those of a conventional enterprise. Using the theory of fuzzy mathematics, the fuzzy hybrid assessment embedded nonlinear integer programming model is established. In Section 3, the tabu search algorithm with an embedded fuzzy hybrid assessment is developed for the model. The experimental example and computational results are included in Section 4.The conclusions are given in Section 5.

II. PROBLEM AND MODEL DESCRIPTION

Risk programming problem for a VE can be described as follows. The fuzzy description of each risk factor in each risk, with and without risk control strategies, is assumed to be known. Being dealt with risk control strategies, the risk factor will be controlled by some way. There are some strategies for each risk. The effect of each strategy on the corresponding risk is different; thus, the description for each risk factor and the cost of the different control strategies are different. The aim of risk programming is to minimize the level of global risk by optimally combining these strategies, constrained by the certain risk cost investment. Hence, the model of the risk programming problem is described as follows:

M1:

min (Global risk level) (1)

$$\text{s.t.} \quad \sum_{i=1}^{n}\sum_{j=0}^{J_i} c_{ij}x_{ij} \le S \tag{2}$$

$$\sum_{j=0}^{J_i} x_{ij} = 1 \qquad i = 1,2,3,\ldots n \tag{3}$$

$$x_{ij} = 0 \ or \ 1 \quad i = 1,2,\ldots n \ \ j = 0,1,\ldots,J_i \tag{4}$$

Where

$$x_{ij} = \begin{cases} 1 \ strategy \ \mathrm{j} \ is \ selected \ by \ risk \ i \\ 0 \ strategy \ \mathrm{j} \ is \ not \ selected \ by \ risk \ i \end{cases}$$

Where S is the total cost investment for the risk control; i is the index of the risk; j is the index of the risk control strategy (strategy 0 for risk i means that no strategy is used for this risk);c_{ij} is the cost of strategy j for risk i; n is risk numbers and J_i is available strategy number for risk i.

Considering the project organization mode and the fuzzy characteristics of a VE, the objective of the model M1 is obtained by the fuzzy hybrid assessment hierarchical model(FSEHM)M2 of risk assessment, which is given in Fig.1.In this FSEHM model, each process of the VE project is considered, as well as the fuzzy description of each risk factor. In the FSEHM model, there are 5 levels, from level 0 (top)to level 4(bottom).Level 0,which corresponds to the objective of model M1,is the goal of a VE. Level 1 consists of the sub-goals pursued by the VE, which are different for a special VE. Level 2 consists of the processes, which are the activities in the VE project. Level 3 is made up of the risk events, which will cause the risk in the processes for different sub-goals. Level 4 is the risk factors of the risk event. For a VE, the fuzzy description of these factors with different control strategies is assumed to be known and is described as the degree of membership to each ranking of risk. The risk ranks are given in Table 1.Also, the relative weights of the lower-level factors to the upper-level factors are assumed to be known. It can be seen in Fig.1, that the risk assessment is conducted from the local system (lowest level) to the global system (level 0).The level of fuzzy global risk for certain risk programming solutions can be obtained by the fuzzy hybrid assessment (Lu et al., 1999). The risk at the global level can then be determined according to the rule of the maximum degree of membership according to the level of fuzzy global risk. However, using this method, many solutions may have the same level of global risk, as shown in Fig.2, where d is the risk rank, and α is the membership degree. According to the rule of the maximum degree of membership, the global risk level of the two solutions in Figs.2 (a) and (b) are same, at 4.However, the risk state of Fig.2 (a) is better than that of Fig.2 (b) taking into consideration all of the degrees of membership of these solutions, which cannot be shown according to the rule of the maximum degree of membership. Therefore, considering the membership degree and risk states, the objective is modified as follows:

$$\min\left(\sum_{q=1}^{Q} d_q^p \alpha_q^p(x)\right) \tag{5}$$

Where q is the risk ranking is the number of rankings of risks, and p is an integer larger than 0. However, if p is set to an integer larger than 1 in formula(5),the global risk level of the solutions in Fig.2(a)will be larger than that in Fig.2(b)accordingly, which is not the case.So,1 is selected for p in our analysis.

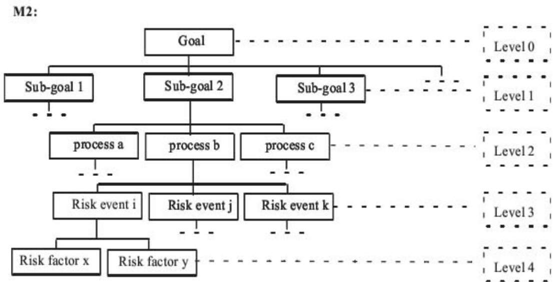

Fig.1. the hierarchical model for evaluating risk.

TABLE I DEFINITION OF RISK RANKS

Risk rank	0	1	2	3	4	5	6	7	8
Risk level	None	Smallest	Smaller	Small	Medium	More	Big	Bigger	Biggest

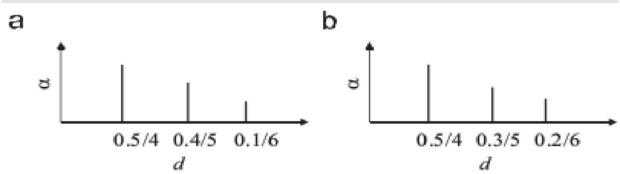

Fig.2. the fuzzy hybrid assessment value of different solutions.

III. ALGORITHM DESIGN

Esq. (1)–(4) is a hybrid of the combination problem and integer programming. The size of the solution space (the number of feasible solutions) of problem (1)–(4) can be determined by the number of risks and a control strategy for the risks, which is $\prod_{i=1}^{n} C_{J_{i+1}}^{1}$ with no constraints. It is easy to see that the problem is NP-hard. Therefore, the heuristic algorithm of this problem is needed for practical use. The local search (LS) algorithm is an effective method for solving large-scale combination optimization problem and integer programming problems. The common heuristic algorithms only adapt to special problems and easily sink into local optimal solutions. The LS algorithm can overcome this shortcoming. It has been used on an increasing number of practical problems and has proven to be effective (Glover, 1989, 1990).

A. *The coding scheme and model transformation*

For our LS, the natural number string is selected as the code description. Let w= {w_1; w2;...;w_n},where w_i is an integer between 0 and J_i, $\forall$ i. This stands for that the control strategy w_i is selected for risk event i. Thus, w= {w1; w2;...; wn} refers to a strategy selection. For example, a natural number string of 10 bits:

{1;3;4;2;2;2;3;3;3;0}

is a strategies selection of 10 risk events. This means that the control strategy 1 for risk event 1, the control strategy 3 for risk event 2, and so on, are selected. It should be noted that bit 10, 0 means that no strategy of risk 10 is selected. Not all of the risks are controlled in a risk programming solution due to limited cost investment in risk control. Objective (1) and constraints (2)–(4) are then equivalent to the following optimization problem:

$$\min\ R(w) = (\sum_{q=1}^{Q} d_q \alpha_q (w)) \tag{6}$$

$$\text{s.t.} \quad \sum_{i=1}^{n} c_{iw_i} \le S \tag{7}$$

$$w_i \in [0,1,...J_i], \quad i = 1,2,...n \tag{8}$$

We can see that the rewritten model (6)–(8) is much simpler than the original model (1)–(4), but that there are some variables in its subscripts. It is difficult to treat this problem using the traditional mathematical model, but easy using a LS.

B. *The initial point*

To generate an initial point, for each risk i, a random number between 0 and J_i is generated. If this point does not satisfy the constraints, a new point is generated randomly again to replace it till the constraints are satisfied.

In order to increase the optimization rate, a multi-point search strategy is used in our LS. The multi-point search strategy will create some initial points each time, which will communicate each other during the search. The optimal point discovered by one point is the forbidden point of the others. This can prevent repeated searching between different points. In order to generate the initial points, the following procedure INI is introduced:

Procedure INI:

Let ps=0 be the counter of the population size, and Pop_size be the population size, then:

Step 0:ps=ps+1,if ps ≤ Pop_size go to Step 1; otherwise,stop;

Step 1:i=0,go to Step 2;

Step 2:i=i+1;if i>n and constraint(7)is satisfied, record the selection and go to Step 0;if i4n and constraint (7)is not satisfied, go to Step 1;if i ≤ n generate a random variable r within the of interval of[0,1],let k=0,and go to Step 3;

Step 3: k=k+1; go to Step 4;

Step 4: if k/ (J_i+1) ≥ r, then let w_i=k-1 and go to Step 2;otherwise, go to Step 3;

From the INI procedure, we can see that the initial points are feasible; as the strategies are selected from the possible strategies and constraints (7) and (8) are considered.

C. *The definition of neighborhood*

Considering the natural number string used in this algorithm, neighbor is defined as all feasible solutions obtained by changing the value of one bite of the current point. That is to say, when the control strategy of one risk is changed, and the new solution satisfies the constraints, it belongs to the neighborhood. Let NB be the set of the neighborhood and w= {w_1;w_2;...;w_n} be the current solution. Then,

$$NB = \{w'/\{w' - w'_k = w - w_k \ \& \ w'_k \ne w_k \ \ for \ \forall k\}, \\ \sum_{i=1}^{n} c_{iw'_i} \le S, w'_k \in [1,2,...J_k]\} \tag{9}$$

For instance, in a natural number string of 5 bits:

{1;2;0;1;3},

the control strategy code corresponding to each risk is

risk 0 {0;1;2;3}
risk 1 {0;1;2;3;4}
risk 2 {0;1;2;3}
risk 3 {0;1;2}
risk 4 {0;1;2;3;4}

Where the underlined number stands for the selected control strategy. If{0,2,0,1,3}satisfies the constraints, then it is the neighbor solution, while only the control strategy of the first risk is changed. Considering the characteristics of our problem, the search tabu list is composed of a two-dimensional integer array. The number of rows is the length of the search tabu list, the first column stands for the code of the risk, and the second column is the code of the control strategy corresponding to the risk of the first column.

$$TS = \begin{bmatrix} 2 & 1 \\ 3 & 2 \\ 1 & 0 \end{bmatrix}$$

If the current point is(1,2,0,1,4,y),then referring to the last row of the search tabu list, we can see that the last movement deletes the point(1,0).Therefore, the former point should be(0,2,0,1,4,y).The search tabu list is renewed according to the criterion of first in, first out.

D. The aspiration criterion

When the optimal point in the neighborhood is worse than the current point and one point in the search tabu list is better than the current point, the tabu is broken.

E. Stopping rule

The rule of maximum searching steps is used as the stopping rule.

F. Fuzzy hybrid assessment(FSE)procedure

According to the risk FSEHM M2 in Fig.1, let L be the level number of the FSEHM model, l be the counter of the level. The goal is on level 0, the sub-goal is on level 1, and so on. Let K_l be the factor number of level l, k be the counter of the factor number, $a_k^l = (a_{k1}^l, a_{k2}^l, ..., a_{kM_{kl}}^l)$ be the weight vector of factors under factor k on level l, M_{kl} be the factor number under factor k on level l, m be the factor counter under factor k on level l,V={v_1;v_2;...;v_n} be the set of the assessment, b_k^l be the fuzzy assessment of factor k on level l, R_k^l be the fuzzy assessment matrix composed of the fuzzy assessment of each factor under factor k on level l. The steps of the fuzzy hybrid assessment are as follows.

Procedure FSE:

Step 1: l=L-2; k is from 1 to k_l; determine R^l_k according to V, then:

$$b_k^l = a_k^l R_k^l = (b_{k1}^l b_{k2}^l, ... b_{kn}^l)(k = 1, 2, ..., K_l) \tag{10}$$

The hybrid assessment of level l is finished; and turns to Step 2.

Step 2: If l=0, stop. Otherwise, let l=l-1, and turn to Step 3.

Step 3: Let k be from 1 to K_l, according to

$b_m^{l+1} = (b_{m1}^{l+1} b_{m2}^{l+1}, ..., b_{mn}^{l+1})(m = 1, 2, ... M_{kl}), R_k^l$ is given as follows:

$$R_k^l = \begin{Bmatrix} b_1^{l+1} \\ b_2^{l+1} \\ . \\ . \\ . \\ b_{M_{kl}}^{l+1} \end{Bmatrix} = \begin{Bmatrix} b_{11}^{l+1} \; b_{12}^{l+1} ... b_{1n}^{l+1} \\ b_{21}^{l+1} \; b_{22}^{l+1} ... b_{2n}^{l+1} \\ ... \\ b_{M_{kl}1}^{l+1} \; b_{M_{kl}2}^{l+1} ... b_{M_{kl}n}^{l+1} \end{Bmatrix} \tag{11}$$

Then, according to Eq.(10)the hybrid assessment of level l is finished. Turns to Step 2. In Eq.(10),the weighted-mean-determining type is used as the compound calculation in the fuzzy metric as shown in Eq.(12).

$$b_{kj}^l = \sum_{i=1}^{M_{kl}} a_{ki}^l b_{ij}^{l+1}; \; j = 1, 2, ... n; k = 1, 2, ..., K_l \tag{12}$$

G. The procedure for the Hybrid Assessment Algorithm based on Local Search

The step-by-step procedure of the Hybrid Assessment Algorithm based on Local Search is as follows:

Procedure FSE-embedded-TS:

Step 1: Specify the parameters:

Pop_size is the number of initial points, Step is the number of search steps.

Step 2: Generate an initial population with Pop_size points using Procedure INI,

$$w(j) = [w_1(j), w_2(j), ... w_n(j)], \; j = 1, 2, ..., Pop\_size.$$

Set the search index to k=0.For point w(j),j=1,2,...,Pop_size, call Procedure OBJ. Then the initial optimal solution w*=w_j*and the optimal objective function value R*=R_j*.

Step 3: Let k=k+1.If k>Step, go to Step 7, otherwise, implement Steps 4–6.

Step 4: For point w(j),j=1;2;...;Pop_size, generate neighborhood NB according to Eq.(9).Go to Step 5.

Step 5:To generate the new population w(j),j= 1,2,...,Pop_size, considering the search tabu list, the long-term search tabu list, the aspiration criterion, and the constraint(7). Go to Step 6.

Step 6: Call Procedure OBJ. If R_{min}<R*,let R*=R_{min} and w*=w(j*).

Step 7: Output R*and w*is the optimal solution.

Procedure OBJ:

Step 1: Call Procedure FSE calculates the global risk level and returns the value of R(j)and the total cost $c(j) = \sum_{i=1}^{n} c_{iw_i(j)}$.

Step 2:Find R_{min}=$\min_{Pop\_size}${R(j)/C(j) ≤ S}.Then, j*=arg{R(j)=R_{min}} is the index of R_{min} that was achieved and the associated control strategy that was selected.

IV. SIMULATION STUDY

A. The example

The first example involves the real-life problem of enterprise bidding for a market opportunity to manufacture lamps. The project consists of five processes. The relationship of processes represented by the Activity-on-Arc mode is shown in Fig.3.The owner has the ability to design the project and carry out core manufacturing, while the bulb manufacturing, cap manufacturing, and assembly processes are finished by partner. The objective of the VE is to minimize the global risk, while the sub-goals are to minimize the cost risk, co-ordination risk, time risk and quality risk, respectively. The weight of each sub-goal to the goal is shown in Table 2.The weight of each process to sub-goal is shown Table 3.The risk events are shown in Table 4.Their relationship to each process under a sub-goal and the weights to the corresponding process are shown in Table 5.Two risk factors, risk probability and risk loss, are considered for each risk event. The weight of the probability and loss to each risk event is shown in Table 4. The fuzzy description of the probability and loss for each risk event when no risk control strategy is used is shown in Table 6.

It can be seen that the example is with 4 sub-goals, 5 sub-processes and 20 risks. The total cost of risk control is 20,000 RMB. The cost of the control strategy for each risk is shown in Table 7, where the highlighted record stands for the selected strategy of optimal solution. The number of the control strategy (including the no control one) for each risk is shown in Table 8.It is clear that there are 96,745,881,600 solutions for this problem without constraints.

TABLE II THE WEIGHT OF THE SUB-GOALS TO THE GOAL

Sub-goal	Cost	Coordination	Time	Quality
Weight	0.3	0.3	0.15	0.25

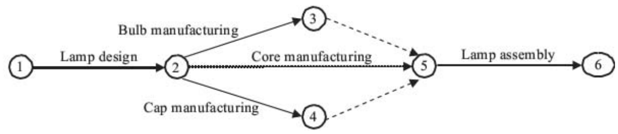

Fig.3.The network for lamp manufacturing

TABLE III THE WEIGHT (W) OF THE PROCESSES TO THE SUB-GOALS

Sub-goal	Process				
	Lamp design	Bulb manufacturing	Core manufacturing	Cap manufacturing	Lamp assembly
Cost	0.5	0.3	0.05	0.15	0
Coordination	0.2	0.5	0.1	0.1	0.1
Time	0	0.2	0.3	0.3	0.2
Quality	0	0	0.4	0.6	0

TABLE IV THE RISK EVENTS AND THEIR PROBABILITY AND LOSS WEIGHT

Risk event no.	Risk event	Risk probability	Risk loss
1	Design method	0.4	0.6
2	Designer's level	0.5	0.5
3	Delay of bulb	0.4	0.6
4	Bad rate of core	0.35	0.65
5	Delay of cap	0.5	0.5
6	Communication with the partner	0.35	0.65
7	Selection of the bulb partner	0.8	0.2
8	Contract award	0.15	0.85
9	Strategy of the partner	0.25	0.75
10	Selection of the cap partner	0.3	0.7
11	Selection of the assembly partner	0.7	0.3
12	The experience of the designer	0.5	0.5
13	The complexity of the product	0.6	0.4
14	Contract management	0.8	0.2
15	The capability of the enterprise	0.3	0.7
16	The experience of the worker	0.7	0.3
17	The capability of the assembly partner	0.5	0.5
18	The reputation of the bulb partner	0.2	0.8
19	The reputation of the cap partner	0.65	0.35
20	The reputation of the assembly partner	0.5	0.5

B. *Parameters setting of LS*

The setting of the values of various parameters in the LS algorithm is important for the efficiency of the algorithm. Therefore, the turning of the parameters of the LS for the problem is presented in this section. The parameters considered are the number of initial points (the population size)‘‘Pop_size’’, the search steps of the algorithm ‘‘Step’’, the length of the search tabu list ‘‘Tabu_size’’, and the length of the long-term tabu list ‘‘Long’’. In the analysis, the performance used to tune the parameters is the ‘‘Best_rate’’, where ‘‘Best’’ stand for the best one of the objective values achieved in 100 runs. The ‘‘Best_rate’’ is the rate to reach the best value. The algorithm was run 100 times with different random seeds for each parameter setting to test the random effect on the solution. Therefore, the parameters with highest “Best_rate’’ are better than others.

(1)The effect of the length of the search tabu list on the “Best_rate”. The analysis on the effect of the length of the search tabu list on the ‘‘Best_rate’ ’is shown in Table 9.The‘‘Cyc_rate’’ is also used in this analysis, where ‘‘Cyc’’ stand for the dead cycle occurred in each of the 100 runs. When the dead cycle occurs, the LS algorithms will loss its “climb” ability. The‘‘Cyc_rate’’ means the rate the dead cycle occurred in 100 runs. It shows that “Best_rate” and “Cyc_rate” vary regularly with the length of the search tabu list. As the length of the search tabu list increases, the ‘‘Cyc_rate’’ decrease, but the ‘‘Best_rate’’ first increases and then decreases. This is because, the longer the search tabu list is, the more points are forbidden and the fewer chances there are to return into the last point. Therefore, the algorithm has a stronger ‘‘climb’’ ability. But, a search tabu list that is too long will cause the “Best_rate” to decline. Obviously, if the algorithm is going to solve the combinatorial optimization problem, the longer the search tabu list is, the more points are forbidden, and the fewer combination alternatives can be chosen; therefore, we have fewer chances to obtain the optimal point. Moreover, the longer the search tabu list is, the lower the ‘‘Cyc_rate’’ is. Considering the columns of ‘‘Cyc_rate’’ and ‘‘Best_rate’’, it is easy to see that if the length of the search tabu list is appropriate, there is less of a chance that the search will go to the dead cycle, and the‘‘Best_rate’’ will be higher.Hence, the dead cycle is one factor affecting the‘‘Best_rate’’, when the search tabu list is of an appropriate length. In order to decrease the‘‘Cyc_rate’’,the long-term search tabu list is used in this research. The long-term search tabu list is able to remember recent moves, and is used to detect whether there is a cycle. If there is, the current point will move to the sub-optimal neighbor solution. Using this method, the cycle can be avoided. However, it is not necessary to detect whether there is a cycle for each step of movement. Therefore, the length of the long-term search tabu list is the other factor affecting the “Best_rate”. Table 9 also shows that the reasonable length of the tabu size is 8.

(2)The effect of the length of the long-term search tabu list on the ‘‘Best_rate’’. The effect of the length of the long-term search tabu list on the ‘‘Best_rate’’ is shown in Table 10.It illustrates that the effect of the long-term search tabu list on the‘‘Best_rate’’ is closely related with the length of the search tabu list. When the search tabu list is of an appropriate length, the long-term search tabu list will have a better effect. In addition, we can see from Table 10 that, using a long-term search tabu list can increase the ‘‘Best_rate’’ by avoiding the cycles, although a long-term search tabu list has an auxiliary effect on the search tabu list. A reasonable length for a long-term search tabu list is 80.

(3)The effect of the size of the population on the “Best_rate”. The effect of the size of the population on the “Best_rate” is shown in Table 11.The results show that one initial point leads to an unfavorable result. This is because some initial points do not lead to the optimal point, but to local optimal ones. Hence, the less random initial point leads to a low ‘‘Best_rate’’. The multi-point search strategy can overcome this shortcoming. As is shown in Table 11,the data suggests that the multi-point search strategy causes an obvious increase in the ‘‘Best_rate’’, and 5 is the suggested population number. A larger population size will only require more search time, but lead to no improvement to the solution.

(4)The effect of the number of search steps on the “Best_rate”. The effect of the number of search steps on the “Best_rate”is shown in Table 12.The data in the table suggests that 1000 is a proper number of search steps for the problem. Less search steps will lead to the decrease of the ‘‘Best_rate’’, however, more search steps will only lead to more time consumed but with no increase of the ‘‘Best_rate’’. The above analysis has shown that the number of initial points ‘‘Pop_size’’=5,the number of search steps ‘‘Step’’=1000,the length of the search tabu list “Tabu_size”=8, and the length of the long-term search tabu list ‘‘Long’’=80 are a reasonable combination of parameters for this problem. The effect of the problem scale on the algorithm is then studied.

(5)The effect of the problem scale on the‘‘Best_rate’’. In order to analyze the effect of the problem scale on the ‘‘Best_rate’’, according to the former example, we reset the problem scale with 5 sub-goals, 5 sub-processes and 30 risks. The control strategy (including the no control one) number for each risk is shown in Table 13.

There are 8,916,100,448,256,000 solutions for this example without constraints, which is 92,160 times more than the first example. Using the reasonable parameter combination, the optimal rate is given in Table 14. As is shown in Table 14, the problem scale grows extremely rapidly with the risk number. The tabu search algorithm can achieve the optimal solution with a higher probability but the computation time does not increase quickly with increase in problem size. In general, the simulation suggests that the tabu search algorithm is effective for this kind of problem.

TABLE V THE RELATIONSHIP AND WEIGHT OF THE RISK EVENT TO THE PROCESSES UNDER THE SUB-GOALS

Sub-goal	Process				
	Lamp design	Bulb manufacturing	Core manufacturing	Cap manufacturing	Lamp assembly
Cost	Risk 1 (0.7) Risk 2 (0.3)	Risk 3 (1.0)	Risk 4 (1.0)	Risk 5 (1.0)	-
Coordination	Risk 6 (1.0)	Risk 7 (0.5) Risk 8 (0.4) Risk 9 (0.1)	-	Risk 10 (1.0)	Risk 11 (1.0)
Time	Risk 12 (0.2) Risk 13 (0.8)	Risk 14 (1.0)	Risk 15 (0.8) Risk 16 (0.2)	-	Risk 17 (1.0)
Quality	-	Risk 18 (1.0)	-	Risk 19 (1.0)	Risk 20 (1.0)

TABLE VI THE FUZZY DESCRIPTION OF THE PROBABILITY AND LOSS FOR EACH RISK EVENT WITH NO RISK CONTROL

Risk event		Risk rank								
		0	1	2	3	4	5	6	7	8
1	Probability	0.0	0.0	0.0	0.3	0.5	0.2	0.0	0.0	0.0
	Loss	0.0	0.0	0.0	0.0	0.2	0.8	0.0	0.0	0.0
2	Probability	0.0	0.0	0.3	0.3	0.3	0.1	0.0	0.0	0.0
	Loss	0.0	0.0	0.0	0.0	0.3	0.5	0.2	0.0	0.0
3	Probability	0.0	0.0	0.1	0.2	0.4	0.3	0.0	0.0	0.0
	Loss	0.0	0.3	0.4	0.3	0.0	0.0	0.0	0.0	0.0
4	Probability	0.0	0.0	0.0	0.0	0.0	0.6	0.4	0.0	0.0
	Loss	0.0	0.0	0.0	0.2	0.6	0.2	0.0	0.0	0.0
5	Probability	0.0	0.0	0.1	0.2	0.4	0.3	0.0	0.0	0.0
	Loss	0.0	0.3	0.4	0.3	0.0	0.0	0.0	0.0	0.0
6	Probability	0.0	0.0	0.0	0.5	0.5	0.0	0.0	0.0	0.0
	Loss	0.0	0.0	0.0	0.5	0.5	0.0	0.0	0.0	0.0
7	Probability	0.0	0.0	0.0	0.3	0.4	0.3	0.0	0.0	0.0
	Loss	0.0	0.0	0.1	0.2	0.3	0.3	0.1	0.0	0.0
8	Probability	0.0	0.0	0.1	0.8	0.1	0.0	0.0	0.0	0.0
	Loss	0.0	0.0	0.1	0.8	0.1	0.0	0.0	0.0	0.0
9	Probability	0.0	0.5	0.5	0.0	0.0	0.0	0.0	0.0	0.0
	Loss	0.0	0.5	0.5	0.0	0.0	0.0	0.0	0.0	0.0
10	Probability	0.0	0.0	0.0	0.0	0.3	0.4	0.3	0.0	0.0
	Loss	0.0	0.8	0.2	0.0	0.0	0.0	0.0	0.0	0.0
11	Probability	0.0	0.1	0.3	0.5	0.1	0.0	0.0	0.0	0.0
	Loss	0.0	0.0	0.0	0.4	0.4	0.2	0.0	0.0	0.0
12	Probability	0.0	0.0	0.5	0.5	0.0	0.0	0.0	0.0	0.0
	Loss	0.0	0.0	0.0	0.3	0.5	0.2	0.0	0.0	0.0
13	Probability	0.0	0.0	0.4	0.3	0.3	0.0	0.0	0.0	0.0
	Loss	0.0	0.0	0.6	0.2	0.2	0.0	0.0	0.0	0.0
Table 6 (*continued*)										
14	Probability	0.0	0.0	0.0	0.5	0.5	0.0	0.0	0.0	0.0
	Loss	0.0	0.0	0.0	0.5	0.5	0.0	0.0	0.0	0.0
15	Probability	0.0	0.0	0.4	0.3	0.3	0.0	0.0	0.0	0.0
	Loss	0.0	0.0	0.0	0.4	0.4	0.2	0.0	0.0	0.0
16	Probability	0.0	0.0	0.3	0.3	0.3	0.1	0.0	0.0	0.0
	Loss	0.0	0.0	0.0	0.0	0.3	0.5	0.2	0.0	0.0
17	Probability	0.0	0.0	0.0	0.0	0.0	0.2	0.4	0.4	0.0
	Loss	0.0	0.0	0.5	0.5	0.0	0.0	0.0	0.0	0.0
18	Probability	0.0	0.0	0.3	0.3	0.2	0.2	0.0	0.0	0.0
	Loss	0.0	0.0	0.0	0.2	0.2	0.2	0.4	0.0	0.0
19	Probability	0.0	0.0	0.0	0.0	0.0	0.6	0.4	0.0.	0.0
	Loss	0.0	0.0	0.00	0.2	0.6	0.2	0.0	0.0	0.0
20	Probability	0.0	0.8	0.2	0.0	0.0	0.0	0.0	0.0	0.0
	Loss	0.0	0.0	0.0	0.0	0.3	0.5	0.2	0.0	0.0

TABLE VII CONTROL STRATEGY COST FOR EACH RISK

Risk	Strategy	Cost
Risk 1	Strategy 1	1000
	Strategy 2	2000
Risk 2	Strategy 1	3000
	Strategy 2	5000
	Strategy 3	500
Risk 3	Strategy 1	1000
	Strategy 2	2000
	Strategy 3	3000
Risk 4	Strategy 1	500
	Strategy 2	2000
	Strategy 3	1000
	Strategy 4	3000
Risk 5	Strategy 1	1000
	Strategy 2	2000
	Strategy 3	3000
Risk 6	Strategy 1	1000
	Strategy 2	1000
	Strategy 3	1000
Risk 7	Strategy 1	1000
	Strategy 2	2000
Risk 8	Strategy 1	1000
	Strategy 2	3000
	Strategy 3	2000
Risk 9	Strategy 1	1000
	Strategy 2	3000

Table 7 (*continued*)

Risk	Strategy	Cost
	Strategy 3	2000
Risk 10	Strategy 1	1000
	Strategy 2	2000
Risk 11	Strategy 1	3000
	Strategy 2	5000
Risk 12	Strategy 1	1000
	Strategy 2	2000
Risk 13	Strategy 1	2000
	Strategy 2	3000
Risk 14	Strategy 1	500
	Strategy 2	500
	Strategy 3	500
Risk 15	Strategy 1	5000
	Strategy 2	5000
Risk 16	Strategy 1	2000
	Strategy 2	2000
	Strategy 3	3000
Risk 17	Strategy 1	3000
	Strategy 2	5000
Risk 18	Strategy 1	2000
	Strategy 2	3000
Risk 19	Strategy 1	1000
	Strategy 2	2000

TABLE VIII THE NUMBER OF RISK CONTROL STRATEGY FOR EACH RISK(PROBLEM 1)

Risk	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Strategy number	3	4	4	5	4	4	3	4	4	3	3	3	3	4	3	4	3	3	5	3

TABLE IX THE EFFECT OF THE SEARCH TABU LIST LENGTH ON THE''BEST_RATE''

Pop_size	Step	Tabu_size	Long	Cyc_rate (%)	Best_rate (%)
1	1000	3	No	100	16
1	1000	5	No	84	17
1	1000	7	No	71	15
1	1000	8	No	29	47
1	1000	9	No	72	29
1	1000	10	No	31	41
1	1000	11	No	23	33
1	1000	12	No	7	43
1	1000	13	No	0	35
1	1000	15	No	0	22
1	1000	17	No	0	11

TABLE X THE EFFECT OF THE LENGTH OF A LONG-TERM LIST ON THE ''BEST RATE''

Pop_size	Step	Tabu_size	Long	Best_rate (%)
1	1000	3	No	16
1	1000	3	10	16
1	1000	3	20	16
1	1000	3	30	16
1	1000	3	40	16
1	1000	7	No	15
1	1000	7	10	27
1	1000	7	20	23
1	1000	7	30	23
1	1000	7	40	24
1	1000	8	No	47
1	1000	8	10	49
1	1000	8	20	48
1	1000	8	30	48
1	1000	8	60	49
1	1000	8	80	48
1	1000	8	100	47
1	1000	9	No	29
1	1000	9	10	44
1	1000	9	20	40
1	1000	9	30	41
1	1000	9	60	37
1	1000	9	80	37
1	1000	9	100	31

TABLE XI THE EFFECT OF THE NUMBER OF INITIAL POINTS ON THE''BEST_RATE''

Pop_size	Step	Tabu_size	Long	Best_rate (%)
1	1000	8	80	48
5	1000	8	80	95

TABLE XII THE EFFECT OF THE NUMBER OF SEARCH STEPS ON THE ''BEST RATE''

Pop_size	Step	Tabu_size	Long	Best_rate (%)
5	300	8	80	86
5	500	8	80	90
5	1000	8	80	95
5	1500	8	80	95

TABLE XIII THE NUMBER OF RISK CONTROL STRATEGIES FOR EACH RISK (PROBLEM 2)

Risk	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Strategy number	4	4	5	5	3	4	3	3	3	4	3	3	5	4	3
Risk	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Strategy number	3	2	2	4	4	4	3	2	3	4	4	4	4	3	3

TABLE XIV THE EFFECT OF THE PROBLEM SCALE ON THE BEST_RATE

Problem scale	Pop_size	Step	Tabu_size	Long	Best_rate(%)	CPU time (s)
96,745,881,600	5	1000	8	80	95	5.7
8,916,100,448,256,000	5	1000	8	80	93	6.5

V. CONCLUSIONS

Risk programming is a problem inherent in VEs. Minimizing total risk within the risk investment is the key to ensuring the success of the VE. This paper introduced a description of the risk programming problem of VEs.

A Hybrid Assessment Algorithm based on Local Search for this problem was proposed. It has better hybrid performance in terms of both computation speed and optimality efficient. The simulation results suggest that it has the potential to solve practical risk programming problems in VE. In general, the proposed model and algorithm has the potential to be an efficient quantitative tool for risk management in the virtual global business environment.

REFERENCES

[1] Ayyub B.M., "Risk Analysis in Engineering and Economics," *Chapman& Hall/CRC,* Boca Raton, FL. 2003

[2] Bier V. M., Haimes Y. Y., Lambert J. H., Matalas N. C., Zimmerman R., "Survey of approaches for assessing and managing the risk of extremes," *Risk Analysis*, vol.19, p p.83–94, January 1999.

[3] Bernus P., Nemes L., "Organizational design: Dynamically creating and sustaining integrated virtual enterprises," *Proceedings of IFAC World Congress*, vol. A. Elsevier, London. 1999.

[4] Fan M., Lin N. P., Sheu C., "Choosing a project risk-handling strategy: An analytical model," *International Journal of Production Economics*, vol.112, pp.700–713, February 2008.

[5] Glover F., "Tabu search: Part I. ORSA," *Journal on Computing*, vol.1, pp.190–206, March 1989.

[6] Glover,F., "Tabusearch: PartII.ORSA," *Journal on Computing* ,vol. 2, p.p. 4–32. Janurary 1990.

[7] Haimes Y. Y., *Risk Modeling, Assessment, and Management second*, Wiley, NewYork, 2004.

[8] Hallikas J., Karvonen I., Pulkkinen U., Virolainen V., "Risk management processing supplier networks," *International Journal of Production Economics*, vol. 90 ,pp.47–58,January 2004.

[9] W. H. Huang, M. Yung, K. L. Wang, "Solution for a risk-based partner selection problem in a virtual enterprise," *International Journal of Computers and Operations Research*, vol. 30, pp. 213–231, February 2003.

[10] Kliem L., Ludin I. S., *Reducing Project Risk*. Gower Publishing Limited, England,1997

[11] Lu R. S., Lo S. L., Hu J.Y., "Analysis of reservoir water quality using fuzzy synthetic evaluation," *Stochastic Environmental Research and Risk Assessment*, vol. 13, pp. 327–336, May 1999.

[12] Makatsoris C., Leach N. P., Richards H. D., Ristic M., Besant C. B., "Addressing the planning and control gaps in semiconductor virtual enterprises," *Proceedings of the Conference on Integration in Manufacturing*, Galway, Ireland, pp.117–129. 1996

[13] Park K.H., Favrel J., "Virtual Enterprise information system and Networking solution," *Computers and Industrial Engineering*, vol. 37, pp. 441–444, January 1999.

[14] Treleven M. S., Schweikhart S. B., "A risk/benefit analysis of sourcing strategies single vs. multiple sourcing," *Journal of Operations Management*, vol. 7, pp. 93–114, April 1988.

[15] Wang S., Tang X., "Research on risk and income for diversified operation of virtual enterprise," *Operations Research and Management Science*, vol. 11, pp. 1–4, April 2002.

Deng Wei, School of Science & Arts, Shanghai Maritime University, Shanghai City, P. R. China, P.C. 200135
Email: yunfuhuanghe@126.com

Personal Details & Education:
Born Date: March, 1978
Nationality: Hubei Province, China
Major: Management Science and Engineering
M.S. 2001.9- 2004.4, Master of Science, School of Science, Beijing Institute of Technology, Beijing, China
B.S.1997.9- 2001.7, Bachelor Degree of Science, Department of Applied Mathematics, Hubei Normal University, Hubei, China
Research Interests & Experience:
·Operations and Research
·Fuzzy Sets and Application
·Supply Chain Management
·

Neng Wang, School of Economics and Management, Dezhou University, Dezhou, Shandong Province, China, P.C. 253023
Email: nengwang234@163.com

Personal Details & Education:
Born Date: September, 1971
Nationality: Shandong Province, China
Major: Computer Science and Application
Ph.D.2005.9- 2008.7 PhD in Management, Management School of Tianjin University, Tianjin, China
M.E. 2002.9- 2005.7 Master of Engineering, Management School of Tianjin University, Tianjin , China
B.E.1989.9- 1993.7, Bachelor Degree of Engineering, School of control science and engineering, Shandong University, Jinan, China
Working experience:
Associate Professor 1993.9- present School of Economics and Management, Dezhou University, Dezhou, Shandong Province, China
Research Interests & Experience:
·Virtual Enterprise
·Optimization theory and algorithms
·Data Mining
·Supply Chain Management