文章编号: 1000-6788(2010)09-1665-07 中图分类号: C393 文献标志码: A

基于 DSM 返工风险评价矩阵的项目优化与仿真

杨 青, 吕杰峰

(北京科技大学 经济管理学院, 北京 100083)

摘 要 针对项目中活动返工的不确定性, 由返工概率矩阵和返工影响强度矩阵共同构成了 DSM 返工风险评价矩阵, 提出了基于返工风险的遗传算法 (Rework risk GA, RRGA), 同时考虑返工风险、反馈个数、反馈距离及活动的时间和费用建立了 DSM 优化目标函数, 改进了传统遗传算法的变异算子和变异概率. 通过对典型案例进行优化及其结果的仿真, 验证了 RRGA 算法可明显降低返工导致的项目费用和时间, 并降低其结果的波动性.

关键词 项目管理; 设计结构矩阵 (DSM); 返工风险评价矩阵; 遗传算法优化

Project optimization and simulation based on DSM rework risk evaluation matrix

YANG Qing, LÜ Jie-feng

(School of Economics and Management, Beijing University of Science & Technology, Beijing 100083, China)

Abstract To describe the uncertainty of activity rework in design structure matrix (DSM), rework probability matrix and rework impact matrix are presented to constitute the rework risk evaluation matrix. This paper proposes rework risk genetic algorithms (RRGA) which combined with the number and distance of feedback and the cost and time of activity. The traditional GA is improved using multi-point mutation operator and self-adaptive mutation probability. Through optimizing by RRGA and simulating for an example, the result verifies the RRGA can reduce the time and cost of project and their variation.

Keywords project management; design structure matrix; rework risk evaluation matrix; genetic algorithm

1 引言

复杂项目存在大量的信息流, 信息流按其传递方向可分为两类: 一种是与现有活动排序方向一致的顺序流动; 另一种是与现有活动排序相反的逆向流动, 这种信息的逆向流动称为迭代 (Iteration) 或反馈. 过多的反馈会导致项目进度延长、费用增加, 因此必须对迭代和返工进行有效的管理. 传统方法不能很好地描述活动之间的反馈和迭代, 设计结构矩阵 (Design structure matrix, DSM) 能够清晰地反映出项目活动之间的信息交流关系, 有效地分析活动之间的反馈和返工, 可应用于项目管理、项目计划等领域[1−3], 它是一种重要的精益研发工具.

传统的 DSM 优化目标函数主要考虑最小化反馈个数和反馈距离, Steward[4] 提出最小化 DSM 矩阵中上对角线部分的反馈个数; Gebala[5] 提出将上对角线反馈标记尽量靠近对角线; Todd[6] 提出尽量把 DSM 矩阵中的反馈标记 “推” 向矩阵的左下角; 还有一些目标函数是同时最小化反馈个数和距离[7]. 它们的主要不足为: 1) 未考虑到返工的不确定性; 2) 未考虑到活动的费用、时间对优化结果的影响.

本文提出了一种基于 DSM 返工风险矩阵的遗传算法 (RRGA) 求解项目活动排序优化问题.

收稿日期: 2009-06-25
资助项目: 国家自然科学基金 (60672185)
作者简介: 杨青 (1968–), 男, 陕西泾阳人, 副教授, 研究方向: 项目管理、精益管理和优化理论, E-mail: yqbuaa@sina.com.

2 构建基于 DSM 的返工风险评价矩阵

2.1 采用 DSM 矩阵描述信息传递的不确定性

如图 1 中, 传统的 DSM 矩阵常采用信息传递标记 "•" 或 "×" 代替图中的数值, 其含义为: 下对角线的标记表示上游活动完成后信息传递给下游活动, 如 B 完成之后信息传递给 C; 上对角线的标记表示下游活动完成后信息反馈给上游活动, 如 C 完成后信息反馈给上游活动 A, 导致 A 返工. 传统 DSM 矩阵采用确定性的方式描述活动之间的依赖关系, 即要么肯定存在返工、要么肯定不存在返工.

但是, 实际上活动之间的依赖关系常常存在不确定性. 一方面, 活动返工情况的发生具有不确定性, 某活动完工后信息迭代给上游活动, 可能导致上游活动发生返工, 也可能不会发生返工, 即返工以概率的形式存在; 另一方面, 若返工发生, 返工的工作量也具有不确定性, 有可能导致完全返工, 也有可能仅使其部分工作发生返工, 返工的影响强度为 0–100% 之间的数值. 因此, 使用 $[0,1]$ 之间的数表示 DSM 活动之间不确定的藕合关系更符合实际情况 [8].

项目管理中, 风险 (R) 指损失发生的不稳定性, 它由不确定性事件发生的概率 (P) 与其导致的影响 (I) 所决定, 即某事件的风险 $R = P \times I$.

为此, 本文提出由 DSM 返工概率矩阵和返工影响强度矩阵计算返工风险, 并以此作为 DSM 优化目标函数, 建立了基于返工风险的遗传算法 (Rework risk GA, RRGA).

2.2 DSM 返工风险评价矩阵

本文将 DSM 返工概率矩阵和返工影响强度矩阵统称为 DSM 返工风险评价矩阵.

1) 返工概率 (Rework probability, RP) 矩阵. 该 DSM 矩阵描述返工发生的概率. Browning[1] 提出用 $[0,1]$ 之间数值型 DSM 矩阵来表示活动之间引发返工的概率, 非对角线元素 $RP(i,j)$ 的值表示返工的概率. 上对角线元素的数值表示迭代的概率 (一次返工), 下对角线元素表示由于上游活动而引起二次返工的概率. 如图 1(a) 所示, 上对角线的元素 $RP(1,3) = 0.5$ 表示活动 C 完工后导致 A 返工的概率为 0.5, 下对角线元素 $RP(8,4) = 0.3$ 表示活动 D 返工完成后, 它导致 H 活动返工概率为 0.3.

2) 本文引入返工影响强度 (Rework impact, RI) 矩阵, 它描述若返工发生, 返工工作量占该活动总工作量的百分比. 它反映了返工工作量的不确定性. 非对角线元素 $RI(i,j)$ 表示任务 j 导致任务 i 返工时, 任务 i 返工量占该任务工作量的百分比. 如图 1(b) 所示, $RI(1,3) = 0.6$ 表示活动 C 的信息迭代给 A 活动, 导致活动 A 的返工量占其总工作量的 60%.

	A	B	C	D	E	F	G	H	I	J	K	L
A	A		.5									
B		B										
C		.2	C									
D				D	.6	.7						.8
E					E	.5		.4			.7	
F		.3				F						.5
G		.3					G				.8	
H	.2			.3				H	.5		.6	
I			.2			.4			I	.5		
J		.2	.3			.2				J	.6	.8
K		.3	.2								K	
L	.4								.2	.3	.4	L

(a) DSM返工概率矩阵

	A	B	C	D	E	F	G	H	I	J	K	L
A	A		.6									
B		B										
C		.4	C									
D				D	.5	.8						.6
E					E	.3		.5			.8	
F		.7				F						.6
G		.3					G				.8	
H	.7			.5				H	.6		.5	
I			.5			.8			I	.4		
J		.4	.2			.7				J	.5	.3
K		.3	.5								K	
L	.8								.4	.5	.5	L

(b) 返工影响强度矩阵

图 1

3 基于返工风险的遗传算法 (Rework risk GA, RRGA)

3.1 建立基于返工风险的 DSM 优化目标函数

由上述返工概率矩阵和返工影响矩阵建立 DSM 优化目标函数, 目标函数称为总协调成本 TCC (Total coordination cost), 可表示为:

$$TCC = \omega_{NC} \times NC + \omega_{RCC} \times RCC + \omega_{RCT} \times RCT \tag{1}$$

其中, NC 表示由于反馈个数和反馈距离所引起的协调成本; RCC(Rework coordination cost) 表示返工协调成本; RCT(Rework coordination time) 表示返工协调时间; ω_{NC}、ω_{RCC}、ω_{RCT} 分别表示 NC、RCC 和 RCT 的权重系数; ω_{NC}、ω_{RCC} 和 ω_{RCT} 的取值范围均为 $0-1$, 且满足 $\omega_{NC}+\omega_{RCC}+\omega_{RCT}=1$.

与传统的 DSM 优化目标函数类似, NC 表达式为:

$$NC=\omega_n\times\sum_{i=1}^{n}\sum_{j=i+1}^{n}DSM(i,j)+\omega_{ij}\times\sum_{i=1}^{n}\sum_{j=i+1}^{n}(DSM(i,j)\times(j-i)) \tag{2}$$

其中, ω_n 和 ω_{ij} 表示反馈个数和反馈距离的权重系数, 满足 $\omega_n+\omega_{ij}$=1; $(j-i)$ 是 DSM 矩阵中第 i 行、第 j 列的活动到对角线的距离; $DSM(i,j)$ 表示活动 i 和 j 之间交互作用的值 (取值为 0 或 1), n 为活动总数.

返工协调成本 RCC 计算公式为 (3), 它表示考虑返工风险情况下, 与返工相关活动所引起费用增加的累计值. 它反映了活动的费用对项目排序优化的影响. 返工协调时间 RCT 计算公式为 (4), 它表示考虑返工风险情况下, 与返工相关活动所引起时间增加的累计值. 它反映了活动的时间对项目排序优化的影响.

$$RCC=\sum_{i=1}^{n}\sum_{j=i+1}^{n}(RP(i,j)\times RI(i,j)\times\sum_{u=i}^{n}(Cost_u\times RP(u,i)\times RI(u,i))) \tag{3}$$

$$RCT=\sum_{i=1}^{n}\sum_{j=i+1}^{n}(RP(i,j)\times RI(i,j)\times\sum_{u=i}^{n}(Time_u\times RP(u,i)\times RI(u,i))) \tag{4}$$

其中, $Cost_u$、$Time_u$ 表示活动 u 的费用和时间; $RP(i,j)$ 和 $RI(i,j)$ 分别表示返工概率和返工影响强度矩阵中活动 i 和 j 之间交互作用的值.

以图 1 为例 (活动的费用和时间数据见表 1), 得 J 活动返工风险引起的返工协调成本 $RCC_J=(0.6\times0.5+0.8\times0.3)\times(25+124\times0.3\times0.5)=23.54$, J 活动返工风险引起的返工协调时间 $RCT_J=(0.6\times0.5+0.8\times0.3)\times(11+85\times0.3\times0.5)=12.83$.

表 1　项目中各活动的时间与费用

活动	A	B	C	D	E	F	G	H	I	J	K	L
时间	42	38	23	54	12	55	13	83	87	11	13	85
费用	53	42	31	350	15	360	23	102	130	25	23	124

3.2　RRGA 编码方法及选择算子和交叉算子

DSM 的排序优化问题属于组合优化中的 NP-hard 问题, 可采用遗传算法求解[9−10].

1) 编码方式: 本文依据 DSM 矩阵的特点, 采用数组编码方式. 假设项目的活动数为 n, 可采用 n 进制编码, 每个编码位的取值为 $1,2,\cdots,n$, 每个数只能用一次, 基因码的长度为 n. 例如, 对于一个染色体编码 [5-3-4-2-1], 表示从 DSM 的左上到右下的活动排序.

2) 选择算子: 对于每代群体, 计算每个个体的目标函数值 Fi, 然后根据目标函数值计算出每个个体的适应度值 $Fi=100/Fi^2$, 接着对适应度值依据轮盘赌法进行选择.

3) 交叉算子: 本文采用单点交叉算子.

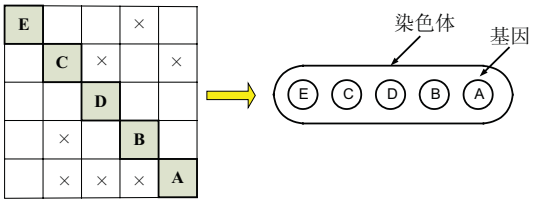
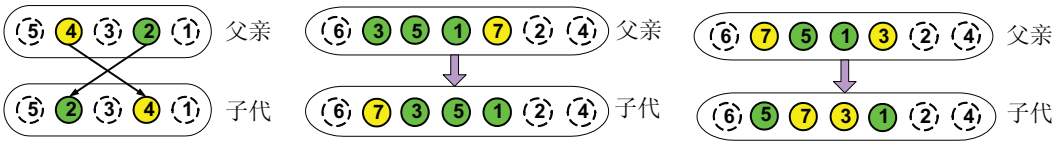


图 2　染色体编码示意图

3.3　变异算子和变异概率设计

由于 DSM 问题本身的复杂程度较高, 对于 n 个活动的 DSM 矩阵, 其可行解空间为 n. 因此, DSM 优化时表现出具有较强的模式欺骗性, 变异操作通过突变的方式产生新的个体, 有利于保持种群多样性, 防止算法的早熟. 如图 3 所示, 典型的变异算子为单点变异算子. 此外, Meier[7] 针对 DSM 的特点提出了 shift 变异算子, 为进一步增强种群的多样性以防止算法的早熟, 本文提出了多点变异 (Multi-point) 算子, 其原理为: 随机选取四个个体, 每组两个个体, 然后两两交换位置.



(a) 单点变异算子 (b) Shift变异算子 (c) 多点(Multi-point)变异算子

图 3 各种变异算子示意图

以图 1 问题为例, 目标函数取式 (2), 种群规模 300, 交叉率 $Pc = 0.9$, 变异概率 $Pm = 0.1$, 分别采用上述三种不同变异算子, 典型的进化过程如图 4 所示. 可见, 多点变异算子的收敛效果明显优于其他两种算子, 收敛的速度也相对较快. 因此, 本文采用多点变异算子.

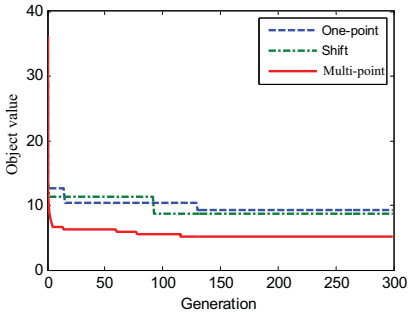


图 4 不同变异算子进化过程

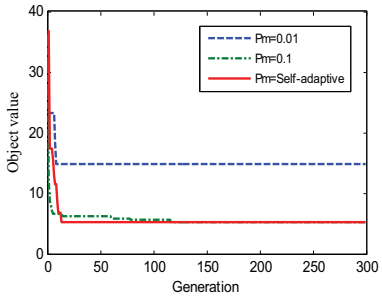


图 5 不同变异概率进化过程

合适的变异概率对于保持种群的多样性同样具有重要作用, 变异概率低不利于保持种群多样性; 变异概率过高可能导致 GA 退化为随机搜索, 收敛速度降低. 本文依据种群适应度均方差与适应度均值之比度量种群多样性, 以此自适应控制变异概率, 降低了计算量[11].

$$p_m = \begin{cases} p_{m,1}, & \text{若 } t \le \lambda T, \text{ 且 } \sigma/\overline{f} < k \\ p_{m,1} \times \dfrac{f_{\max} - \overline{f}}{f_{\max} - f_{\min}}, & \text{若 } t \le \lambda T, \text{ 且 } \sigma/\overline{f} \ge k \\ p_{m,2}, & \text{若 } \lambda T < t \le \beta T \\ p_{m,3}, & \text{若 } \beta T < t \le T \end{cases} \tag{5}$$

式中, k 为常数, λ, β 为给定的百分比, T 为总的进化代数, σ 为当前代种群中个体适应度的均方差, $f_{\max}$、$f_{\min}$, $\overline{f}$ 分别为当前代种群个体适应度的最大值、最小值和均值, $p_{m,1}$、$p_{m,2}$、$p_{m,3}$ 为给定的变异概率.

以图 1 的 DSM 为例, 采用多点变异算子, Pm 分别取 0.01、0.1 和自适应变异概率, 其它参数同上, 得到当前代最优目标值的进化过程如图 5 所示. 可见, 当取自适应性变异概率时, 不仅寻优速度快, 寻优效果也优于固定变异概率.

4 基于返工风险的 DSM 仿真计算流程

由于用返工概率和影响强度描述不确定, 可采用仿真方法进一步分析项目总费用和总时间的分布[12−13].

首先, 规定 DSM 返工中的学习曲线效应, 它是指在产品开发过程中随着迭代次数的增加, 对于同样的工作所花费的时间会逐渐减少, 本文假定曲线为阶段函数[12], 每迭代一次工作效率就提高一个等级. 另外, 为防止仿真过程中迭代无限次地重复, 本文规定最大返工次数取 10.

由优化得到的 DSM 风险评价矩阵, 计算项目总费用和总时间仿真过程如下:

1) 初始化项目输入变量和各状态变量, 完工向量中各活动的完工量为 0;

2) 根据项目活动排序顺序及完工向量, 寻找当前可执行或正在执行的活动集合;

3) 并行执行可执行活动直到所有活动执行完毕, 将这些活动当前时间内的费用和时间加到项目总时间和总费用中;

4) 根据返工概率矩阵和返工影响矩阵, 确定哪些会活动返工及返工的工作量; 考虑学习效应, 修正返工工作量; 更新所有活动的可执行状态以及完工向量;

5) 如果所有的活动都已经完工, 则结束该次仿真, 否则转步骤 2).

执行多次这样的仿真, 当项目总时间和费用达到下面公式的条件时, 说明仿真结果趋于稳定, 停止仿真程序.

$$\frac{|E[TT]_r - E[TT]_{r-25}|}{E[TT]_{r-25}} < 0.005 \tag{6}$$

$$\frac{|\sigma^2[TT]_r - \sigma^2[TT]_{r-25}|}{\sigma^2[TT]_{r-25}} < 0.005 \tag{7}$$

其中 $E[TT]_r$ 为前 r 次仿真得到的项目总时间期望值, σ^2 表示方差, 项目费用的方程与上式类似. 每次仿真过程相互独立, 由仿真计算可以得到项目总费用和总时间的分布情况.

5 案例: 项目活动排序优化及仿真分析

5.1 背景

某企业的 ×× 研发项目包含 12 个活动, 各项设计活动所需的时间和费用如表 1 所示 (为便于计算和比较, 单位取无量纲). 通过专家调查及访谈, 得到项目的排序及返工概率矩阵和返工影响强度矩阵如图 1 所示.

5.2 采用传统目标函数的优化结果及仿真

采用传统 DSM 优化目标函数, 取式 (2), 即不考虑返工的不确定性, 目标函数为同时最小化反馈个数和反馈距离. 其中, ω_n 和 ω_{ij} 分别取 0.7 和 0.3, 遗传算法的参数设置分别为: 种群个数 60, 遗传代数 200, 交叉概率 $Pc = 0.9$, 变异概率 $Pm = 0.1$. 运用 C 语言编程, 得到优化后的活动排序如图 6(a) 所示.

由图 6(a) 所示活动排序及返工概率和返工影响强度, 进行 500 次仿真计算, 得项目总时间、费用均值和方差如表 2 所示, 项目总费用/总时间的联合概率分布图如图 7(a) 所示.

表 2　项目的初始状态及采用不同优化目标得到的结果

序号	优化目标	RCT	RCC	反馈点数	反馈距离	费用均值	费用方差	时间均值	时间方差
1	优化前	238	781	14	41	2283	540	381	62
2	传统目标函数优化结果	67	274	4	8	1613	368	378	39
3	采用 RRGA 优化结果	48	133	4	8	1404	217	339	37

由表 2、图 6(a) 和图 1 可知, 对项目初始状态的活动排序优化后, 反馈个数和反馈距离均有显著的减少, 说明该算法能有效减少信息的反馈, 并因此降低项目总费用和时间.

5.3 采用基于返工风险遗传算法 (RRGA) 的优化结果及仿真

采用 RRGA 算法, 目标函数取式 (1), 其中各参数为: ω_{NC}=0.4, ω_{RCC}=0.4, ω_{RCT}=0.2, 变异概率为自适应变异概率, 其它参数同上. 算法的进化过程略, 优化后的 DSM 矩阵如图 6(b) 所示.

	B	C	A	F	K	L	J	I	E	D	G	H
B	B											
C	.2	C										
A		.5	A									
F	.3			F		.5						
K	.3	.2			K							
L			.4		.4	L	.3	.2				
J	.2	.3		.2	.6	.8	J					
I		.2		.4			.5	I				
E				.5	.7				E			.4
D				.7		.8			.6	D		
G	.3				.8						G	
H			.2		.6			.5		.3		H

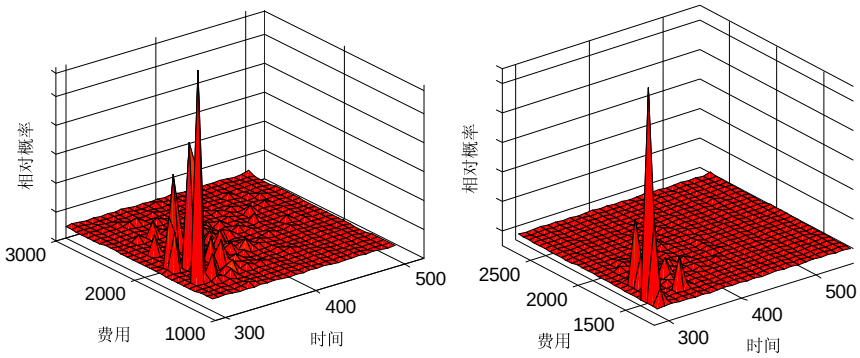
(a) 传统目标函数优化结果

	B	C	A	K	L	F	J	G	I	E	H	D
B	B											
C	.2	C										
A		.5	A									
K	.3	.2		K								
L			.4	.4	L		.3		.2			
F	.3				.5	F						
J	.2	.3		.6	.8	.2	J					
G	.3			.8				G				
I		.2			.4	.5			I			
E				.7		.5				E	.4	
H			.2	.6					.5		H	.3
D					.6	.7				.6		D

(b) RRGA优化结果

图 6　优化后的返工概率矩阵

将优化结果执行 500 次仿真, 计算结果见表 2, 项目总费用/总时间的联合概率分布图如图 7(b) 所示. 对比图 7(a) 和图 7(b) 可知, 采用 RRGA 后项目总费用和时间的分散程度进一步降低. 由项目总费用与总时间的累计概率图 (图 8 所示), 可得项目在不同费用和不同时间完成的概率.



(a) 传统目标函数优化结果 (b) RRGA 优化结果

图 7 优化后的项目总费用/时间联合概率分布图

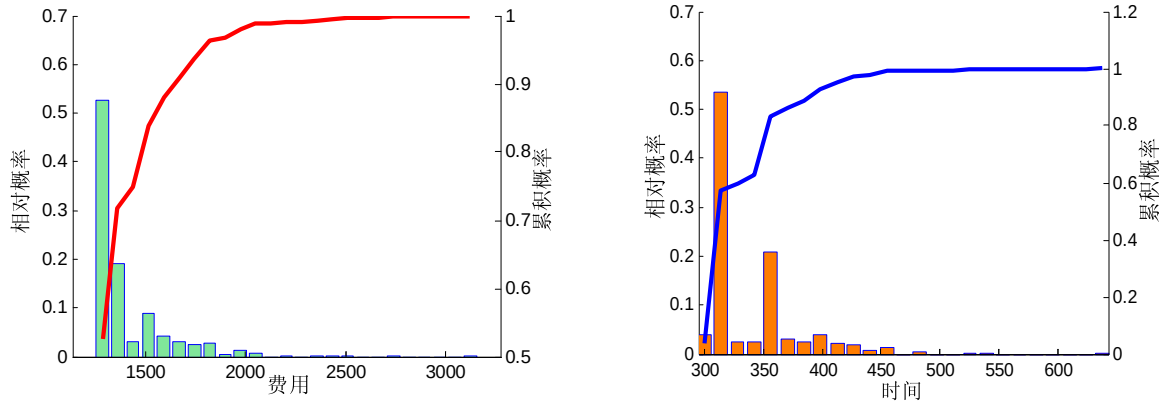


图 8 RRGA 优化后项目总费用与时间累积概率曲线

比较分析上述计算结果, 可得如下结论:

1) 采用 RRGA 算法可明显降低返工导致的项目费用和时间. 由图 1 和表 2 可见, 优化前的 DSM 矩阵存在大量反馈, 采用 RRGA 优化后反馈减少, 项目总费用和时间均值分别降低了 38.5%和 11.0%. 与传统优化方法相比, RRGA 优化后结果得到了进一步优化.

2) 由于 RRGA 算法考虑了概率因素, 因此其优化结果显示数据的波动性得到了降低. 由图 7, RRGA 优化后项目费用和时间的方差有较大的降低, 数据更趋稳定和集中, 图 8 显示累积概率曲线在开始也较陡, 同样说明数据较集中.

3) 传统 DSM 目标函数只考虑反馈量, 而 RRGA 能够依据活动时间、费用和返工风险等因素区别对待各活动, 费用较大的活动将被 “尽量” 避免处于返工循环中, 从而能减少项目总费用和返工协调成本. 以图 6 中费用较大的活动 D 和 F 为例, D 活动在传统优化方法后的返工风险为 $(0.4\times0.6)\times(0.5\times0.5)=0.06$, RRGA 优化后无返工; 同理, 活动 F 返工风险也得到了较大程度的降低, 进而会降低项目的返工协调成本和总费用.

4) 由于返工风险考虑了返工的不确定性, 采用仿真计算技术分析优化得到的结果, 可更直观地显示项目费用和时间的分布, 仿真技术为此类问题提供了可视化分析方法.

6 结论

针对传统的 DSM 优化方法过于强调反馈个数和反馈距离以及未考虑返工风险的不足, 本文由返工概率矩阵和返工影响强度矩阵共同构成返工风险评价矩阵, 并提出了 DSM 基于返工风险的遗传算法 (RRGA). 在遗传算法中引入多点变异算子和自适应变异概率, 以提高算法的全局收敛性. 典型案例的仿真计算验证了 RRGA 算法的有效性.

参考文献

[1] Browning T R. Applying the design structure matrix to system decomposition and integration problems: A

review and new directions[J]. IEEE Transactions on Engineering Management, 2001(3): 292–306.
[2] Chen C H, Ling S F, Chen W. Project scheduling for collaborative product development using DSM[J]. International Journal of Project Management, 2002, 21: 291–299.
[3] 白思俊, 万小兵. 基于设计结构矩阵的项目进度周期 [J]. 系统工程理论与实践, 2008, 28(11): 51–54.
Bai S J, Wan X B. Project schedule cycle based on design structure matrix[J]. Systems Engineering — Theory & Practice, 2008, 28(11): 51–54.
[4] Steward Donald V. The design structure system: A method for managing the design of complex systems[J]. IEEE Transactions on Engineering Management, 1981, 28(3): 71–74.
[5] Gebala D A, Eppinger S D. Methods for analyzing design procedures[C]// Proceedings of the ASME Third International Conference on Design Theory and Methodology, Miami, FL, 1991.
[6] Todd D. Multiple criteria genetic algorithms in engineering design and operation[D]. UK: Engineering Design Centre, University of Newcastle upon Tyne, 1997.
[7] Kusiak A, Larson N. Decomposition and representation methods in mechanical design[J]. Journal of Mechanical Design, 1995, 117(3): 17–24.
[8] Yassine A A. An introduction to modeling and analyzing complex product development process using the design structure matrix (DSM) method[J]. Quaderni di Management (Italian Management Review), 2004(9): 8–9.
[9] Meier C, Yassine A, Browning T R. Design process sequencing with competent genetic algorithms[J]. Transactions of ASME, 2007, 129(6): 556–558.
[10] 盛海涛, 魏法杰. 基于遗传算法的设计结构矩阵优化方法研究 [J]. 中国管理科学, 2007, 15(4): 98–103.
Sheng H T, Wei F J. The research and application of genetic-based design structure matrix optimization algorithm[J]. Chinese Journal of Management Science, 2007, 15(4): 98–103.
[11] 杨青, 邱菀华. 基于遗传算法的制造流程价值优化 [J]. 系统工程学报, 2005, 20(5): 524–529.
Yang Q, Qiu W H. Manufacture process value optimization based on genetic algorithms[J]. Journal of Systems Engineering, 2005, 20(5): 524–529.
[12] Browning T R. Modeling and analyzing cost, schedule, and performance in complex system product development[D]. Massachusetts Institute of Technology, 1998.
[13] Cho S H, Eppinger S D. A simulation-based process model for managing complex design projects[J]. IEEE Transactions on Engineering Management, 2005, 52(3): 316–328.