Computer Modelling and New Technologies, 2013, vol. 17, no. 3, 27–34
Transport and Telecommunication Institute, Lomonosov 1, LV-1019, Riga, Latvia

OPTIMAL BUDGET REASSIGNMENT PROBLEM AMONG SEVERAL PROJECTS WITH DIFFERENT PRIORITIES

D. Greenberg

Department of Economics and Business Administration, Faculty of Social Science
Ariel University of Samaria, Ariel, Israel
dorongreen@gmail.com

The problem of optimal planning for a design office comprising PERT-COST network projects with different priorities is solved. At the upper level – the design office level – the problem centres on reassigning the total budget in order to optimize the combination of projects' reliabilities and priority values. At the second level – the project level – the problem's solution boils down to optimal budget reassignment among the project's activities subject to the least permissible project's reliability value. The solution is obtained by a combination of heuristic and Monte-Carlo methods.

Keywords: budget reassignment, project's reliability, PERT-COST project, global search method, Monte-Carlo method, project's design office.

1. Introduction

This paper is actually a continuation of paper [6] where the problem centres on determining the minimal budget value assigned to a PERT-COST project as well as local budget values assigned to the project's activities. In the paper under consideration a hierarchical optimal planning model comprising two levels, is outlined. Only planning techniques are considered. Unlike papers [2–5], we will implement at each hierarchical level, Monte-Carlo methods.

2. Notation

Let us introduce the following terms:

C	-	total design office budget (pregiven);
$G_k(N,A)$	-	the k -th stochastic network project (graph) of PERT-COST type, $k=1,2,...,n$;
n	-	the number of projects;
D_k	-	the due date of the k -th project (pregiven);
η_k	-	priority value of the k -th project (pregiven); note that if $G_{k_1}(N,A)$ is of higher importance than $G_{k_2}(N,A)$, relation $\eta_{k_1} > \eta_{k_2}$ holds;
C_k	-	budget assigned to project $G_k(N,A)$ (to be optimized);
p_k^*	-	the minimal acceptable project's $G_k(N,A)$ reliability (pregiven);
T_k	-	random duration of project $G_k(N,A)$;
$(i,j)_k$	-	activity entering $G_k(N,A)$;
$c(i,j)_k$	-	budget assigned to $(i,j)_k$ (to be determined and optimized);
$t(i,j)_k$	-	random duration of $(i,j)_k$;
$c_{\min}(i,j)_k$	-	the minimal possible budget to operate activity $(i,j)_k$ (pregiven);
$c_{\max}(i,j)_k$	-	the maximal possible budget to operate activity $(i,j)_k$ (pregiven);
$R_k(C_k)$	-	the maximal reliability value $\Pr\{T_k < T_k \mid C_k\}$ on the basis of assigned budget C_k;
$\vec{X}$	-	search point vector;
δC_k	-	search step value for the local budget C_k;
δc_d	-	search step value for the d -th coordinate, $d=1,2,..., M_k$;

M_k	-	the dimension of the Euclidean search space (number of activities $(i,j)_k$);
Δ	-	the pregiven minimal accuracy of the local search process;
γ	-	a random M_k-dimensional value uniformly distributed on a unity simplex;
$p\{t(i,j)_k, c(i,j)$	-	thep.d.f. of $t(i,j)_k$ on condition that budget $c(i,j)_k$ is assigned to $(i,j)_k$. Note that $R\{C_k\} = \max_{\{c(i,j)_k\}} \Pr\left\{T_k \le T_k \middle| C_k, \sum_{\{(i,j)_k\}} c(i,j)_k = C_k\right\}$ subject to $\sum_{\{(i,j)_k\}} c(i,j)_k = C_k$, $c_{\min}(i,j)_k \le c(i,j)_k \le c_{\max}(i,j)_k$, $(i,j)_k \subset G_k(N,A)$, holds;
Y_1	-	the pregiven number of consecutive unsuccessful random steps undertaken from a routine initial search point;
Y_2	-	the pregiven number of simulating initial search points.

Note, in conclusion, that similar to [5–6], for each activity $(i,j)_k$ budget $c(i,j)_k$ assigned to that activity enters parametrically in the corresponding p.d.f. $p\{t(i,j)_k, c(i,j)_k\}$.

3. The General Problem

The problem centres on determining:

- budget values C_k^*, $k = 1,2,...,n$, assigned to all projects $G_k(N,A)$,
- local budget values $c^*(i,j)_k$ assigned to activities $(i,j)_k \subset G_k(N,A)$,

– in order to maximize objective

$$W = \max \sum_{k=1}^{n} R_k\left(C_k^*\right) \cdot \eta_k \tag{1}$$

subject to

$$\sum_{k=1}^{n} C_k^* = C\ , \tag{2}$$

$$R_k\left(C_k^*\right) \ge p_k^*\ , \tag{3}$$

$$c_{\min}(i,j)_k \le c^*(i,j)_k \le c_{\max}(i,j)_k\ . \tag{4}$$

Problem (1–4) is a sophisticated and complicated problem which can be solved by a combination of heuristic and simulative methods.

4. Auxiliary Problem I

In [6] an auxiliary Problem I is formulated.

Given $G_k(N,A)$, D_k, $(i,j)_k \subset G_k(N,A)$, $c_{\min}(i,j)_k$, $c_{\max}(i,j)_k$, $p\{t(i,j)_k, c(i,j)_k\}$ and C_k, determine local budget values $c^*(i,j)_k$ assigned to all activities $(i,j)_k \subset G_k(N,A)$, in order to maximize the project's reliability, i.e., determine

$$R(C_k) = \max_{c(i,j)_k} \Pr\left\{T_k \le D_k \middle| C_k, \sum_{\{(i,j)_k\}} c(i,j)_k = C_k\right\} \tag{5}$$

subject to (2, 4).

Two different algorithms of solving Problem I can be used, namely:

- the algorithm based on heuristic procedures;
- the Monte-Carlo algorithm.

In order to simplify the algorithms and taking into account that both algorithms refer to single projects, we will omit further on index *k*.

5. Heuristic algorithm (Algorithm I)

The steps of the well-known classical heuristic algorithm [2–4] are as follows:

Step 1. By any means reassign C among all activities (i, j) entering $G(N, A)$ subject to (2, 4). Note that C exceeds $\sum_{\{(i,j)\}} c_{\min}(i, j)$, otherwise project's reliability R equals zero. Thus, it is always possible to undertake a feasible, non-optimal distribution. In case $C \ge \sum_{\{(i,j)\}} c_{\max}(i, j)$ values $c^{opt}(i, j) = c_{\max}(i, j)$.

Step 2. Implement $c(i, j)$ obtained at Step 1 into the given p.d.f. $p\{t(i, j), c(i, j)\}$ for all activities $(i, j) \subset G(N, A)$.

Step 3. Simulate values $t(i, j)$ with p.d.f. obtained at Step 2, $(i, j) \subset G(N, A)$.

Step 4. Calculate the critical path length of the project $L_{cr}(G, \{t(i, j)\})$.

Step 5. Determine all activities $(i, j) \subset G(N, A)$, which belong to the critical path.

Step 6. Repeat Steps 2–5 N times in order to obtain a representative statistics.

Step 7. Calculate ratio $N'/N = R^{(q)}$, where N' is the number of simulated values $L_{cr}(G(N, A), \{t(i, j)\})$ which do not exceed the due date D, and q is the number of the current iteration.

Step 8. Compare two adjacent ratios $R^{(q)}$ and $R^{(q-1)}$. If both relations

$$R^{(q)} > R^{(q-1)}, \tag{6}$$

$$\frac{R^{(q)} - R^{(q-1)}}{R^{(q-1)}} \ge \Delta \tag{7}$$

hold (value Δ is externally given), apply the next step. If relation (6) does not hold, go to Step 14, with values $c(i, j)$ obtained at the $(q-1)$-th iteration. If relation (6) holds but (7) do not hold, proceed to Step 14 with $c(i, j)$ obtained at the q -th iteration.

Step 9. Calculate frequency $\overline{p}(i, j)$ of each activity (i, j) to be on the critical path (using Step5 for N simulations).

Step 10. Reschedule all the activities in a descending order of values

$$\nu(i, j) = \overline{p}(i, j) \cdot \mu(i, j, c(i, j)), \tag{8}$$

where μ is the average value for the p.d.f. $p\{t(i, j), c(i, j)\}$.

For activities with $\overline{p}(i, j) = 0$ reschedule those activities in descending order of their average values $\mu(i, j, c(i, j))$.

Step 11. Determine activity (i_ξ, j_ξ) *with the highest priority* for which relation

$$Z_1 = c_{\max}(i_\xi, j_\xi) - c(i_\xi, j_\xi) > 0 \tag{9}$$

holds. Activity (i_ξ, j_ξ) is placed at the beginning of the sequence and refers usually to critical activities.

Step 12. Determine activity (i_η, j_η) *with the lowest priority* for which relation

$$Z_2 = c(i_\eta, j_\eta) - c_{\min}(i_\eta, j_\eta) > 0 \tag{10}$$

holds. Activity (i_η, j_η) is placed at the end of the sequence and is usually a non-critical activity, which practically has no influence on the project's reliability.

Step 13. Reassign cost value $Z = \min(Z_1, Z_2)$ from activity (i_η, j_η) to activity (i_ξ, j_ξ). Return to Step2.

Step 14. The algorithm terminates. Values $c(i, j)$ obtained after decision-making at Step 7 determine the maximal project's reliability $R(C)$.

Extensive experimentation shows [2–4, 7] that the outlined above Algorithm I is less efficient than the Monte-Carlo algorithm, especially in cases of relatively small amounts of projects entering the design office. Therefore for both hierarchical levels – the project and the company level – we will use the Monte-Carlo approach.

6. Monte-Carlo algorithm (Algorithm II)

The algorithm's novel step-wise structure is as follows:

Step 1. Generate by means of Monte-Carlo method [1] M independent random varieties with p.d.f. $p(x)=e^{-x}$; let them be $\alpha_1, \alpha_2, ..., \alpha_M$. Here M is the number of activities (i, j) entering project $G(N, A)$.

Step 2. Calculate random varieties $\beta_1, ..., \beta_M$, where $\beta_d = \frac{\alpha_d}{\sum_{d=1}^{M} \alpha_d}$. It can be proven that vector $\vec{\beta}^* = (\beta_1, ..., \beta_M)$ is distributed uniformly on simplex $\sum_{d=1}^{M} \beta_d = 1$, $0 < \beta_d < 1$, $d = 1, 2, ..., M$.

Step 3. Repeat Steps $1 \rightarrow 2$ to obtain another independent vector $\vec{\beta}^{**}$.

Step 4. Calculate vector $\vec{\gamma} = \vec{\beta}^{**} - \vec{\beta}^*$. The latter satisfies $\sum_{d=1}^{M} \gamma_d = 0$, $-1 < \gamma_d < 1$, $d = 1, 2, ..., M$.

Step 5. Enumerate all activities (i, j) entering the project $G(N, A)$, by different ordinal numbers from 1 to M. Denote the d-th activity by a_d, it's corresponding minimal and maximal local budgets by $c_{\min d}$ and $c_{\max d}$, and the budget to be assigned and determined in the course of solving the general problem – by c_d, $d = 1, 2, ..., M$.

Step 6. Simulate by means of Monte-Carlo the initial search point $\vec{c}_d^{\,0}$

$$\vec{c}_d^{\,0} = \vec{c}_{\min d} + \vec{b}_d, \tag{11}$$

where

$$b_d = (c_{\max d} - c_{\min d}) \beta_d^* \cdot \Theta, \tag{12}$$

$$\Theta = \frac{C - \sum_{d=1}^{M} c_{\min d}}{\sum_{d=1}^{M} \left[(c_{\max d} - c_{\min d}) \beta_d^* \right]}. \tag{13}$$

Step 7. In case when for some activities a_d $c_d^0 > c_{\max d}$, set $c_d^0 = c_{\max d}$.

Step 8. Similarly to Step 10 of Algorithm I, reschedule all activities in the descending order of their contribution to the project's reliability.

Step 9. Calculate $C' = \sum_{d=1}^{M} c_d^0$. If $C' > C$, calculate the difference $C' - C$ and diminish the budget of non-critical activities in order to equalize the summarized decrease to that difference. The process of diminishing the budget starts from below, i.e., from the least important activities.
In case $C' < C$ the calculated difference $C - C'$ has to be spent on increasing the budget of critical activities, starting from above according to the previously rescheduled activity sequence.
Denote by c_d^0 the finally obtained budget levels, $d = 1, 2, ..., M$.

Step 10. Solve the classical simulation problem to simulate reliability value $R(C)=\Pr\{T_{cr}<D/C, c(i,j), c_{\min}(i,j), c_{\max}(i,j)\}$. The problem can be solved by determining p.d.f. $p\{t(i,j), c^0(i,j)\leftrightarrow c_d^0\}$ for all activities entering the project and later on, by simulating values $t(i,j)$, $(i,j)\subset G(N,A)$. Afterwards, by repeating Step 10 numerous times (similar to Steps 2–7 of Algorithm I), reliability value $R(C)$ is determined. Call it henceforth value R^0, i.e., the initial search point's reliability.

Step 11. Undertake a local search from the initial point

$$\vec{X}^{(0)}+\Delta\vec{X} \Rightarrow \vec{X}^{(1)}, \tag{14}$$

where ΔX is the random search increment.

Here $\vec{X}^{(0)}=\vec{c}^{\,0}$ and $\Delta\vec{X}=\vec{\gamma}\cdot\delta c_d$, where δc_d is the d-th coordinate's search step and $\vec{\gamma}_d$ has been obtained at Step 4.

Step 12. In the course of carrying out the local random search, we will use the optimum trial random search algorithm [1]. Step 12 centres on considering a sequence of Q independent M-dimensional increments $\Delta\vec{X}_q$, $q=1,2,...,Q$, with q vectors $\vec{\gamma}_q=(\gamma_{q1},\gamma_{q2},...,\gamma_{qM})$ satisfying $\sum_{d=1}^{M}\gamma_{qd}=0$. After implementing each q-th increment a correction of coordinate values is undertaken as follows: if $c_d^{(1)}>c_{\max d}$ set $c_d^{(1)}=c_{\max d}$ and in case $c_d^{(1)}<c_{\min d}$ set $c_d^{(1)}=c_{\min d}$.

Later on an additional correction for each q-th increment $\Delta\vec{X}_q$ has to be undertaken similar to that described at Step 9.

Step 13. At each search point $\vec{X}_q^{(1)}$, $q=1,2,...,Q$, Step 10 is applied to calculate the project's reliability $R(\vec{X}_q^{(1)})$. Take the search point with the *maximal* value $R(\vec{X}_\xi^{(1)})$, $1\le\xi\le Q$. If $R(\vec{X}_\xi^{(1)})$ exceeds $R(\vec{X}^0)$, point $\vec{X}_\xi^{(1)}$ is chosen as the new initial search point, i.e., $\vec{X}_\xi^{(1)}\Rightarrow\vec{X}^{(0)}$. Go to the next step.

If $R(\vec{X}_\xi^{(1)})$ does not exceed $R(\vec{X}^0)$ the search terminates at point $\vec{X}^0$.

Note that in the course of undertaking a local random search vector $\vec{\gamma}$ is always renewed by operating Steps 1–4.

Step 14. Check relation

$$\frac{R(\vec{X}_\xi^{(1)})-R(\vec{X}^{(0)})}{R(\vec{X}^{(0)})} \ge \Delta. \tag{15}$$

If (15) holds, return to Step 11. Otherwise apply the next step.

Step 15. Store the results of the local search process [vector $\vec{c}_d$, $R(\vec{X}^{(1)})$] in a special array.

Step 16. Counter f of the number of simulating initial points $\vec{X}^0$ works, $f:=f+1$.

Step 17. If $f\le Y$, return to Step 1. Otherwise apply the next step.

Step 18. Take the *maximal* project's reliability of the Y initial search points stored in a special array (see Step 15). The corresponding reliability $R(C)$ together with the optimal vector $\vec{c}_d$, is the solution of Problem I.

7. Auxiliary Problem II

The problem is as follows:

For a single PERT-COST project $G(N,A)$ with given values $c_{\min}(i,j)$, $c_{\max}(i,j)$, $(i,j)\subset G(N,A)$ and p^*, determine the *minimal* budget C assigned to that project, together with values $c(i,j)$, which enables

$$R(C) \geq p^* . \quad (16)$$

Thus, the problem's formalization is as follows:

$$\min C \quad (17)$$

subject to

$$R(C) = \Pr\{T < D/C\} \geq p^* \quad (18)$$

and (2, 4).

The enlarged step-wise algorithm to solve Problem II is as follows:

Step 1. Set $C = \sum_{\{(i,j)\}} c_{\min}(i,j)$.

Step 2. Set counter $h := 1$.

Step 3. Calculate $C := C + h \cdot \delta C$.

Step 4. Solve auxiliary Problem I to obtain $R(C)$.

Step 5. Compare $R(C_G)$ and p^*. If $R(C_G) \geq p^*$ holds, proceed to Step 7. Otherwise apply the next step.

Step 6. Counter h works, $h := h + 1$. Return to Step 3.

Step 7. The algorithm terminates with the minimal budget value obtained at Step 3 and values $c(i,j)$ determined by solving auxiliary Problem I at Step 4.

Note that value C obtained at Step 1 enables solving Problem B at Step 4. However, the corresponding initial reliability value $R(C)$ will be extremely small.

8. The General Problem (GP)

The general idea of solving GP (1–4) is based on implementing the Monte-Carlo method and is very similar to the algorithm outlined in Section 6. The enlarged algorithm's step-wise structure is as follows:

Step 0. Given (see Notation):

- total budget value C;
- local projects $G_k(N,A)$, $k = 1,2,...,n$;
- priority values η_k, $k = 1,2,...,n$;
- due date values D_k, $k = 1,2,...,n$;
- values $c_{\min}(i,j)_k$ and $c_{\max}(i,j)_k$, $k = 1,2,...,n$, $(i,j)_k \subset G_k(N,A)$;
- minimal projects' reliability values p_k^*.

Step 1. Solve separately for each project $G_k(N,A)$ auxiliary Problem II. Determine the minimal budget values by C'_k, $k = 1,2,...,n$. Thus,

$$C'_k = \min(C_k) \quad (19)$$

subject to

$$R(C_k) \geq p_k^* . \quad (20)$$

Step 2. Check inequality

$$C \geq \sum_{k=1}^{n} C'_k . \quad (21)$$

If the inequality does not hold, the general problem has no solution. Otherwise calculate

$$\Delta C = C - \sum_{k=1}^{n} C'_k .$$

Step 3. Similarly to Steps $1 \to 2$ of Algorithm II for solving auxiliary Problem I, generate n random values $\beta_1^*, \beta_2^*, ..., \beta_n^*$ with vector $\vec{\beta}^*$ being distributed uniformly on simplex $\sum_{k=1}^{n} \beta_k^* = 1$.

Simulate the initial search point $\vec{X}^{(0)}$

Step 4.

$$\vec{X}^{(0)} \equiv \vec{C}_k^{(0)} = \vec{C}_k' + \Delta C \cdot \vec{\beta}_k . \quad (22)$$

Step 5. Calculate objective W for the initial search point

$$W^{(0)} = \sum_{k=1}^{n} R_k\left(C_k^{(0)}\right) \cdot \eta_k , \quad (23)$$

where $R_k\left(C_k^{(0)}\right)$ is calculated by means of solving auxiliary Problem I (either by using Section 5 or Section 6).

Step 6. Similarly to Steps 3 and 4 of Monte-Carlo Algorithm II (see Section 6), calculate vector

$$\vec{\gamma} = \vec{\beta}^{**} - \vec{\beta}^{*} , \quad (24)$$

which satisfies

$$\sum_{k=1}^{n} \gamma_k = 0 , -1 < \gamma_k < 1, \quad k = 1,2,...,n . \quad (25)$$

Step 7. Undertake a routine search step in an n-dimensional space $\vec{X}^{(0)} \Rightarrow \vec{X}^{(1)}$, where $\vec{X}^{(1)}$ is determined by

$$\vec{X}^{(1)} = \vec{X}^{(0)} + \Delta C \cdot \vec{\gamma} . \quad (26)$$

Step 8. If in the course of any search step coordinate k, $k = 1,2,...,n$, satisfies , the search step is regarded non-feasible, and we apply Step 11. Otherwise proceed to the next step.

Step 9. Calculate objective $W^{(1)}$ (see Step 5) for the search point $\vec{X}^{(1)}$ in order to compare values $W^{(0)}$ and $W^{(1)}$.

Step 10. If $W^{(1)} > W^{(0)}$, search point $\vec{X}^{(1)}$ is set as the initial one, $\vec{W}^{(1)} \equiv \vec{W}^{(0)}$; clear counter f and return to Step 6. Otherwise apply the next step.

Step 11. Counter f_1 of the number of consecutive unsuccessful search steps taken from search point $\vec{X}^{(0)}$ works, $f_1 = f_1 + 1$.

Step 12. If $f_1 \le Y_1$, return to Step 6 to simulate the next routine search step to be made from point $\vec{X}^{(0)}$. Otherwise, apply the next step.

Step 13. Counter f_2 of the number of simulated initial search points works, $f_2 = f_2 + 1$.

Step 14. If $f_2 \le Y_2$, return to Step 3. Otherwise, proceed to the next step.

Step 15. Choose the maximal value of objective W from Y_2 initial search points stored in a special array; consider the chosen value to be the optimal value $W^{(opt)}$.

Step 16. For all values $C_k^{(opt)}$ entering objective $W^{(opt)}$ remember all values $c(i, j)_k$ which have been previously determined at Step 5.

Optimal values $c^{(opt)}(i, j)_k$, which together with values C_k have to be stored in a special array, form the general problem's solution.

9. Conclusions

The following conclusions can be drawn from the study:

1. The paper presents a hierarchical budget reassignment model in the form of unification of several single-level models.
2. Single-level models can be optimized by means of Monte-Carlo methods as well as sophisticated heuristic techniques.
3. When the number of projects entering the design office is relatively small we suggest using the Monte-Carlo approach.

Acknowledgement

The author expresses his gratitude to Prof. D. Golenko-Ginzburg for the kind attention and valuable comments during preparation of this work.

References

1. Fishman, G. (1997). *Monte Carlo Concept, Algorithms and Applications*, Springer Series in Operation Research, New-York.
2. Golenko-Ginzburg, D. (1972). *Statistische Methoden der Netzplantecknik*, Leipzig: BSB B.G. TeubnerVerlagsgesellschaft (in German, translated from Russian: *Statistical Methods in Network Planning and Control*, Moscow: Nauka, 1968).
3. Golenko-Ginzburg, D. (2010). *Stochastic Network Modelsin R&D Projecting*, Voronezh: NauchnayaKniga (in Russian).
4. Golenko-Ginzburg, D., Burkov, V. and Ben-Yair, A. (2011). *Planning and Controlling Multilevel Man-Machine Organization Systems under Random Disturbances*, Ariel University Centre of Samaria, Ariel: Elinir Digital Print.
5. Golenko-Ginzburg, D., Greenberg, D. and Ben-Yair, A. (2007). Time-cost optimization problem for deterministic PERT networks, *Communications in DQM*, 10(2): 80–87.
6. Golenko-Ginzburg, D., Ljubkin, S. and Swid, N. (2013). Budget reassignment methods in innovative stochastic network projects, *Journal "Research and Innovations"*, 1 (in print).
7. Malisheva, A. (2005). Control and planning models for aggregated projects in a project office, Ph.D. Thesis, *Ben-Gurion University of the Negev*, Beer-Sheva, Israel.

Received on the 1st of June 2013