Aggregation errors in project scheduling

O A Lyakhov
Institute of Computational Mathematics and Mathematical Geophysics of Siberian Branch of
Russian Academy of Sciences, Ac. Lavrentieva ave., 6, Novosibirsk, 630090, Russia
E-mail: loa@rav.sscc.ru

Abstract. It is shown that aggregation leads to errors in network models projects problems
with restricted resources. In consequence of aggregation an optimal solution of network problem
has systematic mistakes in resources evaluation and project schedules look like better than
indeed. As illustration the results of computer experiments on real-life data are presented.

1. Introduction
Having appeared in the middle of the last century, network planning has not lost its relevance and
is used for creation plans of complex projects with limited resources in the project management.
Network models are the main tool for displaying technological and resource relationships between
operations. All known project management systems, including the most popular Microsoft and
Primavera, are based on network models [1, 2]. At the initial stage in the practical solution
of network planning problems, are used approximate methods, that do not guarantee the
construction of optimal plans (see, for example, [3, 4, 5, 6, 7]). Present period zero-one linea
models for solving small dimension scheduling problems are used [8, 9, 10, 11].

In the article the problem of resources evaluation in network projects models with constant
and variable intensity of operations such as PERT or CPM is discussed. The exactness of plans
depends on a degree of resources aggregation and a time scale, which consists of some intervals
(quanta). A quantum is the least time period (called so by analogy with physics) for measuring
resources and other purposes in scheduling. A sum of quanta is equal to planning period.

Resources demands for schedules are calculated separately per each quantum taken as a
whole. A decrease of a number of quanta (one way of aggregation) reduces models dimensions,
but at the same time it leads to growth of systematic mistakes of resources calculation, which
doesn’t allow to see schedules in proper perspective.

The results of numerical experiments using the package for solving network optimization
problems [12] are presented.

2. Simple example
The schedule of three operations, which use the same resource (each operation requires one unit
of resource at each moment) is given in table 1.

The amount of resource may be used in one-time unit period is equal to 2.
Table 1. The schedule of three operations

| Operations | Timing schedule | Processing times of operations |
|------------|----------------|-------------------------------|
|            | start | finish |                      |
| 1          | 0     | 2      | 2                     |
| 2          | 1     | 3      | 2                     |
| 3          | 1     | 4      | 3                     |

Figure 1. Resource demand for one-time unit period.

Figure 2. Resource demand for two-time unit period.

If the length of a quantum is one time unit, the resource requirements for schedule are the following (figure 1).

The schedule is not feasible: the requirement of resource in quantum 2 (interval [1,2]) is 3, more than may be used.

The same schedule for quanta of length 2 is feasible (figure 2)

By analogy it may be shown that resources amalgamation into bigger groups (second way of aggregation) distorts a value of schedule feasibility.

3. A value of resources requirements mistakes in consequence of aggregation

Let we have an aggregative schedule with the quantum length equal to $k \cdot d$ \textit{(where $k > 1$)}. A resource deficit in one quantum is:

$$\Delta_{kd} = \max(kv - kR, 0),$$

where $kv$ is the resource requirement in given quantum; $kR$ is the amount of resource may be used in one quantum.

If $\Delta_{kd} = 0$, there is enough resource for all operations in given quantum. $\Delta_{kd} > 0$ shows resource deficiency in given quantum.

A resource deficit of the same schedule in the same period (length $k \cdot d$) may be calculated as a sum of deficits of $k$ quanta with the length $d$:

$$\Delta d = \sum_{i=1}^{k} \max((v_i - R), 0),$$

where $\sum_{i=1}^{k} v_i = kv$, $v_i \geq 0$ $i = 1, k$; $v_i$ is a resource requirement in quantum $i$ ($v_i$ are unknown values to the aggregative model).
A mistake of resource calculation in a quantum with the length \(k \cdot d\) in comparison with the quantum length \(d\) is:

\[ m_{kd} = \Delta_d - \Delta_{kd} \]  

(3)

It’s not difficult to show that

\[ \max m_{kd} = \begin{cases} R(k - 1), & \text{if } v > R \\ kv - R, & \text{if } v \leq R \quad \text{and} \quad kv > R \\ 0, & \text{if } v \leq \frac{R}{k} \end{cases} \]  

(4)

A mistake of resource evaluation in one quantum may reach \(|k(v - R)|\). In many resources network models formulas for mistakes of aggregation may be received by summarizing all resource mistakes in all quanta.

3.1. Some conclusions

1. An aggregation (by amalgamation of resources into bigger groups or by increase of quantum length) leads to appearance of systematic mistakes in resources evaluation which reduce real misbalance of production possibilities (because \(m_{kd} \geq 0\)).
2. A mistake value is proportional to a quantum length.
3. A feasible schedule with the quantum length \(k \cdot d\) will be guarantee feasible to the quantum length \(d\), if requirements of each resource in each quantum will not be bigger than \(1/k\) of the available amount of resource in each quantum.

4. Experimental test on mistakes of aggregation in network models

In real-life scheduling there are often used two kinds of network problems:
- to balance timed resources requirements if a completion date is set;
- to minimize overtime of completion date if timed resources requirements are restricted.

The first problem uses criteria minimizing a sum of resources overuse in all quanta. In consequence of aggregation an optimal solution of this problem has systematic mistakes of resources evaluation. Not feasible schedule of the second problem may become feasible because of aggregation. As a result of aggregation solutions of both problems look like better than indeed.

In the next parts of article computer experiments on the first problem resources evaluation mistakes are described.

4.1. The definition of the model

Many networks of projects (jobs, works, orders) model is given. Each project is described by its own network, has its setup beginning and finishing dates (not necessarily). Each operation of each project is characterized by its volume (in time units), number of resource and intensity. A processing time of operation is a variable, which may be determined as a function of intensity, volume and distributed resource. A scheduling period consists of time intervals of the same length (quanta). It needs to find a schedule minimizing a sum of resources overuse at all quanta.

Designations:
\[ G = (E, M) \]  

is the network model (graph with directed arcs);
\( E \) is the set of nodes (operations);
\( M \) is the set of arcs which represent operations connection;
\( \Gamma^{-1}(i) \) is the set of operations (nodes), which are predecessors to \( i \), \( i \in E \) (\( j \in \Gamma^{-1}(i) \) if there exists the arc \((j, i) \in M)\);
$\Gamma^{+1}(i)$ is the set of operations to which $i \in E$ is the predecessor ($j \in \Gamma^{-1}(i)$ if there exists the arc $(i, j) \in M$);

$S_{ki}$ is the amount or resource $k$, required by operation $i$ at one unit of time (intensity);

$t_1(i)$ is start time of operation;

$t_2(i)$ is finish time of operation;

$x = \{(t_1(i), t_2(i))\mid i \in E\}$ is the set of vectors of start and finish operations times (schedule);

$S_{ki}$ is the minimal limit of intensity of $k$-resource usage by operation $i$;

$\overline{S}_{ki}$ is the maximal limit of intensity of $k$-resource usage by operation $i$;

$B_{kl}$ is the amount of resource $k$ available in quantum $l$, $l = 1, T$;

$T$ is the number of quanta in the planning period;

$t_\alpha$ is the start time of planning period;

$t_\omega$ is the finish time of planning period;

$V_{ki}$ is the volume of resource $k$ to proceed operation $i$ (in time units);

$V_{kil}$ is the volume of resource $k$ required by operation $i$ in quantum $l$;

$N_{il}$ is the duration of operation $i$ in quantum $l$.

It needs to find $t_1(i), t_2(i), S_{ki}, \forall i \in E, k = 1, K$, which satisfy conditions (5)-(10).

\begin{align*}
 &t_2(i) \leq t_1(j), \quad j \in \Gamma^{+1}(i), \quad i \in E; \tag{5} \\
 &t_1(i) \geq t_\alpha, \quad \text{for } i \in E \text{ such as } \Gamma^{-1}(i) = \emptyset; \\
 &t_2(i) \leq t_\omega, \quad \text{for } i \in E \text{ such as } \Gamma^{+1}(i) = \emptyset; \\
 &S_{ki} = \frac{V_{ki}}{t_2(i) - t_1(i)}, \quad k = 1, R; \tag{6} \\
 &\overline{S}_{ki} \leq S_{ki} \leq \overline{S}_{ki}, \quad i \in E \\
 &V_{kil} = S_{ki} \cdot N_{il}, \quad k = 1, R, \quad l = 1, T, \quad i \in E; \tag{7} \\
 &N_{il} = \max(0, \min(t_2l, t_2(i)) - \max(t_1l, t_1(i))), \quad i \in E, \quad l = 1, T; \tag{8} \\
 &\sum_{i \in M} V_{kil} = V_{kl}, \quad k = 1, R, \quad l = 1, T \tag{9} \\
 &F_T(x) = \sum_{k=1}^{R} \sum_{l=1}^{T} \max(V_{kil} - B_{kl}, 0) \rightarrow \min \tag{10}
\end{align*}

Conditions (5) determine the structure of network model (such as PERT or CPM) and requirements to complete all operations in the planning period. Formulas (6) connect intensities and processing times of operations. Conditions (7), (8) are used to specify the parts of volumes and processing times of operations in quanta. Group (9) determines resources requirements in quanta. The criteria function (10) defines the total timed deficit of resources.

1 We have many projects with their network models. Consolidation them into one many projects model and adding conditions on setup dates of each project is possible by including dummy nodes, directly connected with nodes $i$, if $\Gamma^{-1}(i) = \emptyset$ (start dates) and with $i$, if $\Gamma^{+1}(i) = \emptyset$ (finish dates).

2 It is implied that intensities do not change during implementation of operations.
Let us estimate the criteria function mistake in consequence of quantum length increase. If a duration of a planning period is fixed a quantum length is inversely proportional to $T$. If $T=1$, the model (5)-(10) converts from scheduling to selecting projects for implementation in planning period. If $T$ increases, plans become more exact but at the same time it leads to a growth of model dimension. For each schedule $x$ it is correctly that $F_k(x) \geq F_q(x)$ if $k > q$.  

Let us designate $x^*_q$ - the optimal schedule of $q$-quanta model. Then $F_q(x^*_q) \geq F_{q-1}(x^*_{q-1})$, i.e. the $q-1$ - quanta model solution determines the low limit of $q$-quanta criteria function. On the contrary, $q$-quanta optimal solution determines the high limit criteria functions with number of quanta lesser than $q$.

An aggregation mistake value may be expressed by the following formula:

$$D_{qk} = F_k(x) - F_q(x), \text{ if } k > q. \quad (11)$$

$D_{qk}$ - shows "invisible" in $k$-quanta model deficit of resources, which becomes "visible" if the number of quanta is equal to $q$. By analogy it may be estimated a value of optimal schedules aggregation mistake:

$$D^*_q = F_k(x^*_q) - F_q(x^*_q) \text{ if } k > q. \quad (12)$$

4.2. Experiments on the model

There was used a real-life data of one technological institute.  

For 4 projects, which were selected for implementation in one year planning period, for 7 restricted resources (specialists in different fields) there were solved two problems using the model (5)-(10):

- the first variant for monthly length of quanta ($k=12$);
- the second variant for half monthly length of quanta ($q = 24$).

The algorithm for solving the problem (5-10) is based on the method of random search [12, 13]. It needs 90 iterations to find the optimal schedule for the first variant, which keeps resources balance (see table 2). The same data , being used in 24-quanta model, didn’t allow to receive no deficit solution. The value of the criteria function was equal to 507.288 time units (see table 3).

The optimal solution of 12-quanta model, being estimated in half monthly quanta, has the mistake (11) 14504.410 time units. This is the reason do not recommend to use this schedule in the real-life control. The comparison between calculated values of 24-quanta and 12-quanta criteria functions at iterations of the first variant (table 2, columns, named $F_{24}(x)$ and $F_{12}(x)$) shows that some intermediate 12-quanta schedules are nearer to optimum, than the optimal 12-quanta model solution. According to table 2 the nearest to optimum in 24-quanta model solution was received at the 6-th iteration. It has the resources deficit 8126.116 time units. After the 6-th iteration the improvement of 12-quanta schedule is accompanied only by deterioration of 24-quanta solution: its criteria function is increasing from 8126.116 up to 14504.410 time units. After the 6-th iteration the improvement of 12-quanta schedule is accompanied only by deterioration of 24-quanta solution: its criteria function is increasing from 8126.116 up to 14504.410 time units.

The solution of the second variant after second iteration with resources deficit 4733.510 time units (see table 3 column, named $F_{24}(x)$) is better than the best solution of the first variant. Because of $F_{24}(x)$ determines a high level of $F_{12}(x)$, a decrease of $F_{24}(x)$ from iteration to iteration makes an interval of the value of $F_{12}(x)$ more short. But in some cases the decrease of $F_{24}(x)$ does not lead to reduction of $F_{12}(x)$: from second up to third iteration (see table 3) $F_{12}(x)$ increases. The optimal schedules aggregation mistake (12) $D_{12,24}^*$ is equal to 507.266 time units.

---

3 Each item of $F_k(x)$ is not lesser than corresponding item of $F_q(x)$. It follows from (1)-(4).

4 More detailed description of primary data is given in [13].

5 The algorithm is described in [12]. Each iteration gives a schedule where conditions (5)-(9) are valid.
Table 2. The first variant (calculations, using 12-quanta model)

| Number of iteration | $F_{12}(x)$ | $F_{24}(x)$ | $D_{12,24}(x)$ |
|---------------------|-------------|-------------|---------------|
| 0                   | 25494.790   | 42063.160   | 16568.370     |
| 1                   | 4384.100    | 17794.740   | 13410.610     |
| 2                   | 3855.500    | 22491.880   | 18636.380     |
| 3                   | 3114.200    | 14328.730   | 11214.530     |
| 4                   | 2714.200    | 9890.286    | 7176.086      |
| 5                   | 500.000     | 8342.116    | 7842.416      |
| 6                   | 500.000     | 8126.970    | 7626.970      |
| 87                  | 200.000     | 8126.970    | 7926.970      |
| 89                  | 100.000     | 8126.970    | 8026.970      |
| 90                  | 0           | 14504.410   | 14504.410     |

Table 3. The second variant (calculations, using 12-quanta model)

| Number of iteration | $F_{24}(x)$ | $F_{12}(x)$ | $F_{24}(x) - F_{12}(x)$ |
|---------------------|-------------|-------------|--------------------------|
| 1                   | 10190.130   | 7082.105    | 3108.025                 |
| 2                   | 4733.510    | 3845.386    | 888.124                  |
| 3                   | 4733.510    | 4016.498    | 717.012                  |
| 6                   | 2538.638    | 1531.229    | 1007.409                 |
| 15                  | 1466.007    | 1248.989    | 217.018                  |
| 1714                | 1420.623    | 1240.994    | 179.629                  |
| > 100000            | 507.288     | 428.621     | 78.667                   |

5. Conclusion

Usually quantum length and number of resources groups in network models (a degree of aggregation) are determined from management traditions and computer possibilities. Yearning to receive real-life schedules leads to reduction of large complex project models dimension. As a result feasible schedules may become unrealistic for implementation.

Mistakes of aggregation are systematic: they smooth resources requirements and give more reduced time values than indeed. It leads to inaccuracy in project plans.

Computer experiments show that mistakes are too crude to ignore them in practice. To be sure that a schedule is realistic is to disaggregate a model and to solve scheduling problem.

Acknowledgments

This work was carried out under state contract with ICMMG SB RAS 0251-2021-0005.

References

[1] Power project success today and tomorrow. Available: http://www.primavera.com/ (last visit 09.06.21)
[2] Microsoft Project. Available: https://products.office.com/ru-ru/project/project-management-software?rtc=1 (last visit 09.06.21)
[3] Kofman A and Debazey G 1968 Network Planning Methods: Application of the System PERT and its Varieties in the Management of Industrial and Research Projects (Moscow: Progress)
[4] Zukhovitsky S I and Radchik I A 1965 Mathematical Methods of Network Planning (Moscow: Nauka)
[5] Golovinsky V 1966 Industrial Scheduling (Moscow: Progress)

6 The large complex projects models system was selected for experiments [13] in the computing center [14]. It was taken into consideration the accessibility of its software and its real-life usage.
[6] Ahuja H N 1976 Construction Performance Control by Networks (New York)

[7] Hartmann S and Briskorn D 2010 European journal Operational Research Review of variants and continuations the problem of project planning with limited resources 207(1) pp 1-14

[8] Pritzker A B, Watters Y L and Wolfe P M 1969 Management science Planning multiple projects with limited resources: zero-one programmatic approach 16 pp 93-108

[9] Kolish R and Sprecher A 1996 European journal of operational research PSPLIB-library project planning 96 pp 205-216

[10] Project scheduling problem library - PSPLIB. Available: http://www.om-db.wi.tum.de/psplib/ (last visit 09.06.21)

[11] Time-constrained project scheduling. Available: https://www.utwente.nl/en/eemcs/tcpsp/ (last visit 09.06.21)

[12] Lyakhov O A 2012 Foundation of algorithms and programs Software package for network planning scheduling (Novosibirsk, ICM&MG SB RAS, PR12002)

[13] Lyakhov O A 1996 Two-level model for complex jobs planning (Novosibirsk Russian academy of science. Siberian branch. Computing center. Preprint 1075)

[14] Mikhailenko B G, Glinskiy B M, Kuchin N V and Chernykh I G 2014 AMCA 14, Novosibirsk, June 8-11, 2014 The Siberian Supercomputer center for collective use: current state and development trends (Novosibirsk: Akademizdat) p 67