Integer linear models of network planning

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

Abstract. The application of integer linear programming for accurate solutions in project management is considered. The proposed network models for building schedules include works (operations) performed continuously and with interrupts. Examples of building optimal makespans using well known software for solving integer programming problems are given.

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 in conditions of limited resources in the operational management. Network models are the main tool for displaying technological and resource relationships between operations. All known project management systems, including the most popular ones from Microsoft and Primavera, are based on network models [1, 2]. At the initial stage in the practical solution of network planning problems, were used approximate methods, that do not guarantee the construction of optimal plans (see, for example, [3, 4, 5, 6]). At present, there is a growing interest in methods for accurately solving on networks scheduling, in particular in integer programming (see, for example, review [7]). It is necessary to describe the conditions of network planning in the form of mathematical programming models to apply integer programming methods.

2. Definitions

The network model of the complex of works (operations) is the oriented without contours and loops graph \( G = (I, M) \).

- \( I \) - is the set of vertices (works of the network model).
- \( M \) - is the set of relationships between works (precedence relations). If \( i, j \in I \) and \( i \) immediately precede \( j \), then \( i \in \Gamma^{-1}(j) \) and the pair \( (i, j) \in M \) is denoted (the execution of operation \( j \) can begin no earlier than all operations \( i \in \Gamma^{-1}(j) \) are completed).

Similarly, if \( j \) immediately follows \( i \), then \( j \in \Gamma^{+1}(i) \) is denoted. The input operation is \( \alpha \in I \), which has no antecedents, i.e. \( \Gamma^{-1}(\alpha) = \emptyset \). The output is \( \omega \), such that \( \Gamma^{+1}(\omega) = \emptyset \). There is at least one input and one output dummy (resource-free) operation.

Each work \( i \in I \) is associated with the volume of the required resource type "power" (non-stackable) \( V_{ik}, i \in I, k = 1, K, K \) – the number of types of resources.

1 Operations can be displayed as arcs.
2 Non-stock or non-accumulated – resources that are lost when they are not used, for example, workers, equipment. In English language literature these resources are called renewable.
The processing time of an operation can be either constant or variable value and depends on the amount of allocated resources. In the first approximation assumes that the limiting for each operation is a single resource and the intensity of resource consumption is constant over the entire interval of the operation: \( t_i = V_{ik}/r_{ik}, \) \( i = 1, N, \) \( t_i \) – processing time of operation; \( r_{ik} \) – operation intensity (volume of resource consumed per unit of time).

It’s need to build a schedule that takes into account resource constraints, satisfying some criteria (for example, minimizing a completion of a project or reduction of unbalance of resources at the specified cycle, etc.).

3. The Model for minimizing the duration of the General cycle when limited resources (constant intensity)

\[ y_{it} \in [0, 1], \text{ integer, } i = 1, N, t = 1, T \]  
\[ y_{it0} \cdot \sum_{t=1}^{t_0} y_{jt} = 0, \forall j \in \Gamma^{+1}(i), i = 1, N, t_0 = 2, T \]

This condition can be written in linear form:

\[ y_{it0} + Q \sum_{j \in \Gamma^{+1}(i)} \sum_{t=1}^{t_0} y_{jt} \leq 1, \ i = 1, N, \ i \neq \omega, \ t_0 = 2, T \]

\[ \sum_{t=1}^{T} y_{it} = t_i, \ i = 1, N \]  
\[ \sum_{i=1}^{N} r_{ik} \cdot y_{it} \leq R_{kt}, \ k = 1, K, t = 1, T \]

\[ t_i = \frac{v_{ik}}{r_{ik}}, \ i = 1, N \]

\[ \sum_{t=1}^{T} y_{\omega t} \cdot C_t \rightarrow \min \]

3.1. Variables

\( y_{it} = 1 \) if the work \( i \) is performed in the \( t \) - th time interval and 0 otherwise.
\( t_i \) – duration of \( i \)-th work \( i \in I, \ i \neq \omega.\)
\( r_{ik} \) – the intensity of the \( i \)-th operation (the amount of consumed resource \( k \) per unit of time).

For each operation, the there is a single renewable resource. In some, the integer condition must be met, for example, indivisible workers or units of equipment.

3.2. Constants

\( t_{\omega} = 1 \) – the duration of the dummy work;
\( T \) – the number of time intervals in the planning period (planning horizon).

The choice Of \( T \). The problem is guaranteed to be joint when \( T = \sum_{i \in I} t_i \). Too much \( T \) increases the dimension , for small \( T \), there may be no valid solutions. Therefore, it is possible to select a large value and then reduce it in the process of solving the problem, or estimation of \( T \) preliminary solution of auxiliary problems.

\( R_{kt} \) – the volume of \( k \)-th avaible renewable resource in \( t \)-th time interval \( k = 1, K, t = 1, T \).
\( Q \) – a pre-defined value determined from the relations \( Q \cdot \max_{i \in I} \left( \sum_{j \in \Gamma^{+1}(i)} \sum_{t=1}^{t_0} y_{jt} \right) \leq 1, \ t_0 = 2, T \), for example, \( Q \leq \frac{1}{T \cdot N} \).

\( C_t \) – constants chosen from the relations \( C_{t+1} > C_t, \ t = 1, T - 1. \)
3.3. Explanation of conditions

Conditions (2) specify the execution of works in sequence, given by network model $G$ (the execution of all directly following the $i$-th operation, can be started after it completions). Linear and nonlinear records of this condition are given. Conditions (3) and (5) guarantee the performance of $i$-th work for the time $t_i$ with intensity $r_{ik}$. Conditions (4) in each time interval do not allow you to use more resources than it is possible.

In the criterion function (6) $\omega$ – the last work of the model, i.e. such that $\Gamma^{+1}(\omega)$ is an empty set. In the selection force $C_t$ minimizing this function reduces the duration of General cycle.

3.4. Remarks

The solution of the problem determines the resource requirements over time and timetable of works.

For each operation, $i$ start and end time $(B_i, E_i)$ defined by the formulas

$$B_i = \min_{y_{it} = 1, t = 1, T} t$$

and

$$E_i = \max_{y_{it} = 1, t = 1, T} (t + 1).$$

There may be interruptions in the performance of work.

If there are several end works in the network model, then adding dummy arcs you can get the only output $\omega$.

A slight change in the conditions allows you to record a more general problem of minimizing the duration of the general cycle with a variable in time performance intensity of works.

Replacing the condition group (5) with $\sum_{k=1}^{T} r_{ik} = V_{ik}$ and by introducing restrictions on the minimum and maximum possible resource consumption intensity $r_{ik}^0 \leq r_{ik} \leq r_{ik}^1$, we get a model in which the intensity of operations changes during its fulfilment and conditions (5) become linear.

Models with interruptions can be converted into models without breaks in which all operations have a single duration. To do this, each work must be presented as a path. Quantity of the arcs in the path is equal to the processing time of operation. In this case, the number of variables is increased to $T \cdot \sum_{i=1}^{N} (t_i - 1)$, but it is possible to use known methods of solution the problems with continuous operations (see, for example, [7, 9, 10]).

4. The Model of resource imbalance minimization for a given planning period

In this model, which is closer to real life, the a predetermined value value of the planned period $T$ is given and it’s need to build a schedule with minimal resource requirements deviations from their availability over time. In this case, there may be several network models (multi-volume, multiproject).

In the model with conditions (1), (2), (3), (5) the restrictions (4) are changed to (7) or (8):

$$\sum_{i=1}^{N} r_{ik} \cdot y_{it} \leq R_{kt} + \Delta_{kt}, \quad k = 1, K, \quad t = 1, T$$

$$R_{kt} - \Delta_{kt} \leq \sum_{i=1}^{N} r_{ik} \cdot y_{it} \leq R_{kt} + \Delta_{kt}, \quad k = 1, K, \quad t = 1, T.$$  

(7) \hspace{1cm} (8)

$\Delta_{kt} \geq 0$, a variable that determines the $k$ imbalance resource in the $t$-th time interval. The criterion function (9) or (10) is written as follows:
\[ \sum_{k=1}^{K} \sum_{t=1}^{T} \Delta_{kt} \rightarrow \min \]  
(9)

\[ \sum_{k=1}^{K} \sum_{t=1}^{T} \Delta_{2kt}^{2} \rightarrow \min \]  
(10)

In addition, the model can be added with restrictions on dates for the execution of some works:

\[ \sum_{t=1}^{T} y_{lt} \cdot C_{t} < C_{t0}. \]  
(11)

Condition (11) formalizes the need to complete the operation \( l \) not later than the beginning of the interval \( t_{0} \). \( C_{t} \) - constants are selected from relations \( C_{t+1} > \sum_{i=1}^{N} C_{t}, t = 0, T - 1 \). Similarly, the start due dates can be taken into account.

5. The linear integer model of scheduling at a predetermined intensities of works

In this problem statement, a predetermined duration and the intensity of operations are assumed, i.e. \( t_{i} \) and \( r_{ik} (\forall i, k) \) are constants. This makes it easier to record conditions and allows to formalize the problem in a linear form.

In the General cycle minimization model with limited resources (conditions (1) – (4) with the criterion function (6)), the constraints (4) become linear. In models for minimizing resource imbalances (conditions (1) – (3), (7) or (8) with the criterion function (9) or (10)), the nonlinearity disappears in restrictions (7), (8).

6. The Linear integer model for minimizing the duration of the project, performed continuously with a predetermined intensity

This statement differs from the previous one by the prohibition of works breaks, and from the well known Johnson’s problem, - a partial order of operations. In this model: \( E_{i} \) - late completion time of operation \( i \), \( B_{i} \) - early start time \( i = 1, N \); \( x_{it} \) - a variable equal to 1 if the work \( i \) completes in \( t \)-th time interval, and equal to 0 otherwise, \( i = 1, N, t = 1, T \). \( B_{i} \) and \( E_{i} \) corresponds to early and late event timing in network planning terminology [4].

\[ \sum_{t=B_{i}+t_{i}}^{E_{i}} x_{it} = 1 \quad x_{it} \text{ - integer}, i = 1, N \]  
(12)

\[ x_{it0} + Q \sum_{j \in \Gamma+1(i)} \sum_{t=B_{j}+t_{j}}^{\min\{t_{0}+t_{j}, E_{j}\}} x_{jt} \leq 1 \quad i = 2, N, t \neq \omega, t_{0} = B_{i}+t_{i}, E_{i} \]  
(13)

\[ \sum_{i=1}^{N} r_{ik} \cdot \sum_{t=t_{0}}^{t_{0}+t_{i}} x_{it} \leq R_{kt0} \quad k = 1, K, t_{0} = 1, T \]  
(14)

\[ \sum_{t=B_{i}+1}^{E_{i}} t \cdot x_{it} \rightarrow \min \]  
(15)

In the model (12-15), the start times of operations are calculated subtracting the duration of operations from the finish moments of operations, conditions (13) specify a partial order of work, resource constraints (14) guarantee for each resource in each moment of time that the need does not exceed the availability. Constant \( Q \) is selected in the same way as in conditions 3.
7. Numerical experiments for cycle duration minimization models

The models (1–4, 6) and (12 – 15) were chosen as the basis for the experiments. Some results are shown in the table.

### Table 1. Some results for PSPLIB examples

| Name of problem | Horizon | Number of rows, columns | Min. make-span | Calculation time (sec.) | Number of rows, columns | Min. make-span | Calculation time (sec.) |
|-----------------|---------|--------------------------|----------------|--------------------------|--------------------------|----------------|--------------------------|
| J301_1          | 80      | 1592 1658 50751          | 43             | 87.33                    | 1620 1485 59987          | 43             | 7.88                     |
| J301_1          | 120     | 2952 2898 171851         | 43             | 6445.91                  | 3008 2685 185726         | 43             | 12498.56                 |
| J901_1          | 75      | 1952 2653 34175          | 67             | 34.25                    | 2115 2235 45207          | 73             | 495.05                   |
| J901_1          | 85      | 2882 3589 68404          | 67             | 13611.50                 | 3106 3186 86970          | 73             | 9250.47                  |
| J1201_1         | 102     | 3814 5041 12577          | 99             | 1.31                     | 3720 4262 12683          | No valid solutions | 31.27                   |

There were used the Cplex optimization software [8] and [9] for preparing data in MPS format. Examples are taken from the set of test cases PSPLIB [11, 12], which contains 480 variants of models for 30, 60, 90 works and 600 – for 120 works. All examples are limited to no more than 4 renewable resources. For each variant from the library the best known solutions of two types are given in [13]: proven, i.e. with proven optimality, and just the best at the present moment.

Calculations were carried out on Intel I5-6500 processor (four cores, 3.2 GHz) in Windows 10 environment except some problems, which were solved on a Siberian Supercomputer Center multiprocessor unit [14]. Duration of calculation depends on the choice of the planning horizon value (T). For T, close to the optimum, the minimum makespan and its optimality is found and proved enough quickly. For preliminary assessment T were applied known approximate solutions of problems.

8. Conclusion

The formalization of the conditions calendar network planning with breaks and without breaks of works (operations) with constant and variable intensity of operations, subject to resource constraints in the form suitable to apply the methods of integer linear programming is proposed. The results of numerical experiments using a well-known package for solving optimization problems [8] are presented.

Experiments have confirmed “the relative weakness” of the integer programming for the exact solution of practical problems. The main constraint is the size of input data. Currently, the construction of real schedules is impossible without using approximate algorithms.
References

[1] Power project success today and tomorrow. Avaiable: http://www.primavera.com/ (last visit 18.03.20)
[2] Microsoft Project. Avaiable: https://products.office.com/ru-ru/project/project-management-software?rtc=1 (last visit 18.03.20)
[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] Zuckovitsky S I and Radchik I A 1965 Mathematical Methods of Network Planning (Moscow: Nauka)
[5] Golovinsky V 1966 Industrial Scheduling (Moscow Progress)
[6] Aluja 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) 1-14
[8] Cplex optimizer. High-performance mathematical programming solver for linear programming, mixed-integer programming and quadratic. Avaiable: https://www.ibm.com/analytics/cplex-optimizer (last visit 18.03.20)
[9] Lyakhov O A 2015 Foundation of algorithms and programs Software package for building integer models for project scheduling (Novosibirsk, ICM&MG SB RAS)
[10] Pritzker A B, Watters Y L and Wolle P M 1969 Management science Planning multiple projects with limited resources: zero-one programmatic approach 16 93-108
[11] Kolish R and Sprecher A 1996 European journal of operational research PSPLIB-library project planning 96 205-216
[12] Project scheduling problem library - PSPLIB. Avaiable: http://www.om-db.wi.tum.de/psplib/ (last visit 18.03.20)
[13] Time-constrained project scheduling. Avaiable: https://www.utwente.nl/en/eemcs/tcsp/ (last visit 18.03.20)
[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