Al-Mustansiriyah Journal of Science
ISSN: 1814-635X (print), ISSN:2521-3520 (online) Volume 31, Issue 4, 2020 DOI: http://doi.org/10.23851/mjs.v31i4.923

Research Article Open Access

Branch and Bound Algorithm and an Improvement for Calculating the Nearest Link of Building a Railway Network

Waleed Mohammed Elaibi*

Department of Statistics, College of Administration and Economics, Mustansiriyah University, Baghdad, IRAQ

*Contact email: waleedassistlec@gmail.com

Article Info

Received
21/10/2020

Accepted
26/10/2020

Published
20/12/2020

ABSTRACT

The importance of branch and bound algorithm is the mathematical improvement to find the value of (X) that Maximize or minimize the objective function within a set of feasible solution, as it is reliable on the efficient evaluation of the bounds of regions or branches of the space of the research, whether they are upper or lower. In this paper discuss five cases with respect to branching decisions based on network solutions to calculate the nearest link with a short time. From the results, bound and branch algorithm can develop and change the obtaining solutions for the five cases under study.

KEYWORDS: Branch and bound algorithm, Tree search algorithm, improvement algorithm, decomposition algorithm, network design problem, Network Project Analyses.

الخلاصة

تكمن أهمية خوارزمية التفرع والتحديد في التحسين الرياضي للعثور على قيمة (X) التي تزيد أو تقلل من دالة الهدف ضمن مجموعة من الحلول الممكنة ، حيث يمكن الاعتماد عليها في التقييم الفعال لحدود المناطق أو الفروع .مساحة البحث ، سواء كانت أعلى أو أقل. ناقش في هذه الورقة خمس حالات تتعلق بقرارات التفرع بناءً على حلول الشبكة لحساب أقرب ارتباط في وقت قصير. من النتائج ، يمكن لخوارزمية التفرع والتحديد تطوير وتغيير الحلول من خلال الحالات الخمس قيد الدراسة.

INTRODUCTION

Network design and location models are often called upon to represent complex issues arising in transportation, logistics, telecommunications, and production planning applications. These models are usually formulated as large-scale, combinatorial mixed-integer programs with complex constraint structures.

Branch and bound algorithm is currently the only general tool available for finding optimal solutions to these difficult formulations. Yet, even with the help of efficient specialized algorithms that compute tight bounds on their optimum values, realistically formulated and dimensioned models require the exploration of such huge search trees that optimal solutions cannot be found but for the simplest instances.

A large number of emerging applications in scientific areas including material sciences, bioinformatics, computer vision, and robotics ask for finding a densely connected subgraph of a given graph which maximizes a measure of correlation among its vertices. This problem, which is known in the literature as the tree search algorithm or branch and bound algorithm, is the subject of study of this article, especially when it comes to the topic of improvement [1].

The paper is organized as follows. We draw the problem network understudy and then the branch and bound algorithm for solving the problem are presented and our computational experience is reported to show the nearest link in network structure.

PROJECT NETWORK

Scheduling of activities and the control of their flows through a production process plays most significant role in any modern manufacturing systems. Project scheduling is a complex decision making problem because of conflicting goals, limited resources and the difficulty in accurately modeling real world problems. Project scheduling is a complex decision making problem because of conflicting goals, limited resources and the difficulty in accurately modeling real world problems.

According to the critical path method, the decision-maker can control the time and cost of the project and improve the efficiency of resource allocation to ensure the project quality. When decision-makers engage in evaluating activity times in connection with project network they tend to give assessments based on their wisdom, professional knowledge, experience and available information

Critical Path Method (CPM) and Project Evaluation and Review Technique (PERT) are widely recognized valuable tools for the planning and scheduling of large projects. In CPM a deterministic data for activity time is used where as in PERT random time data's are employed. However, in reality, due to the uncertainty of information as well the variation of management scenario, it is often difficult to obtain the exact activity time estimates of all activities. Thus, the conventional approaches, both deterministic and random process, tend to be less effective in conveying the imprecision or vagueness nature of the linguistic values. One alternative to handle the uncertainty associated with the processing time is to use fuzzy techniques.

Network optimization is a very popular and frequently applied field among the well-studied areas of operations research. Many practical problems arising in the real life situations can be formulated as network models. Every large project consists of many activities. An important aspect of project management is scheduling the activity time accurately. This is a critical component of project planning as this will be the deadline for the completion of a project [2, 7].

SPECIALIZED BRANCH AND BOUND ALGORITHM

Additional improvements can be obtained by the use of special constraints or dummy links, which take advantage of the specific features of the transmission expansion problem.

The algorithm is an enumerative algorithm based on a binary search tree where the nodes represent candidate problems and the branches correspond to decisions taken so far in a given path. All integer solutions are explicitly or implicitly represented in that tree which guarantees that all optimal solutions will eventually be reached by the algorithm.

The branching operation consists of dividing the current problem (a node in the decision tree) into two descendants (the current node is also called the parent node). The descendant problems are easier to solve than the parent problem since one of its variables is defined in the two descendent nodes; additional constraint on the separation variable is added to each of the descendant problems.

A separation variable is a variable that assumes a non-integer value in the current node and can be fired at the values of either neighboring integer values; in both cases, a new constraint is then added to the current problem (contradictory constraints). The current problem is then excluded from the list of candidate problems and the two descendent problems are added to the list (as a result of the branching operation) [3].

DEFINITION of the SEARCH TREE

Supposing that M represents the group of all the possible networks that are formed by that of all links, L. Which is the tree-root is identified with M. A vertex $M_S$of the family of the tree forms a subset M_S of M, that is, a networks-family defined by three subsets of$L$:

1- The set $L_S^I = \lambda_n$of *links that are included;*
2- The set $L_S^E = \overline{\lambda}_n$ of *links that are excluded;*
3- The set $L_S^U$of *links that are unassigned;*

A search tree is shown in figure 1. At each vertex of this tree, the included links that is included $\lambda_n$and those that are excluded $\overline{\lambda}_n$ are identified. In the similar edges of the search tree, the pivot links are also identified. During the stages of the research, a thorough partition of the set $M$of all possible networks is defined by the terminal vertices, and those that are nonterminal can be disregarded.
The research progressively constructs a tree of the research search via (a) the elimination of terminal vertices if possible, or (b) dividing them if the

ISSN: 1814-635X (print), ISSN:2521-3520 (online) Volume 31, Issue 4, 2020 DOI: http://doi.org/10.23851/mjs.v31i4.923

elimination is impossible. If all the terminal vertices are eliminated, then the search is concluded.

A terminal vertex which represents a network with an objective function has been computed is eliminated. A terminal vertex that represents networks that are infeasible is eliminated. It is easy to check feasibility. A terminal vertex $M_S$contains a network that is feasible if and only if the network M_S^{min}formed by the links in L_S^I is suitable. A terminal vertex identified not to contain an optimal solution is got rid of. Suppose a terminal vertex $M_S$and the network M_S^{max}formed by all the links in $L_S^I \cup L_S^U$. Due to the mono tonicity property of the objective function, $M \in M_S \to C(M) \geq C(M_S^{max})$.

Thus, $C(M_S^{max})$ represents the vertex-lower bound M_S. Consequently, if $C(M_S^{max}) > C^*$, where C^* is the objective function- lowest known value, then the vertex M_S can be got rid of. If it is hoped to obtain *an* optimal network, instead of *all the other* optimal networks, the vertex M_S can be got rid of when $C(M_S^{max}) \geq C^*$ [4].

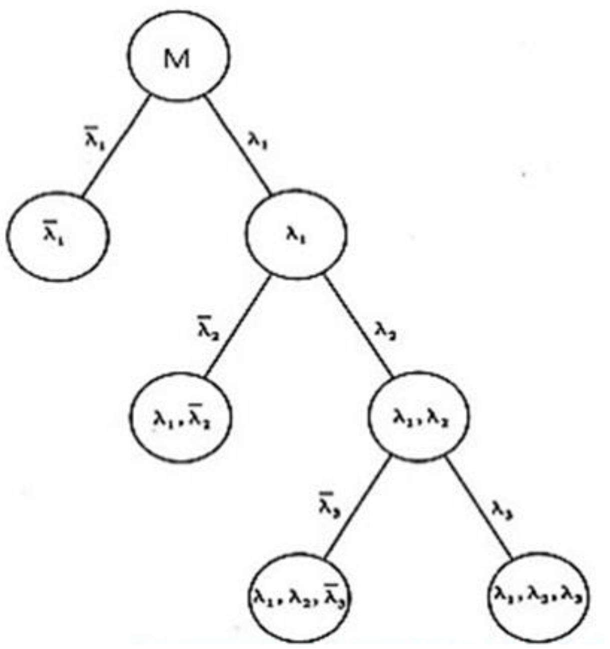

Figure 1. Search tree for three links

Table 1. Crisp Duration for Project.

No	Activities Description	Activity Code	Precedence Activity	Crisp Duration
1	Demolishing concrete structures	A		11.9
2	Take out the buried concrete structures	B		11.9
3	Supplying, pouring and hammering the concrete pillars	C	A	16.2
4	Groundwater disposal works	D	B	15.95
5	Concrete work primary	E	C,D	8.1
6	Moisture-proof	F	C,D	23.55
7	Soil works	G	C,D	23.8
8	Concrete work to build the tunnel	H	C,D	191.9
9	Auxiliary walls	I	G	8.1
10	Water pumps for emergency situations	J	I,F	11.9
11	Installing vacuums (smokers)	K	I,F	8.1
12	Serving ways	L	K	16.45
13	Serving Bridge	M	K	8.35
14	Electrical Works	N	M	8.35
15	Pillars Testing	O	N,H,E,J,L	4.05

Draw the problem network under study:

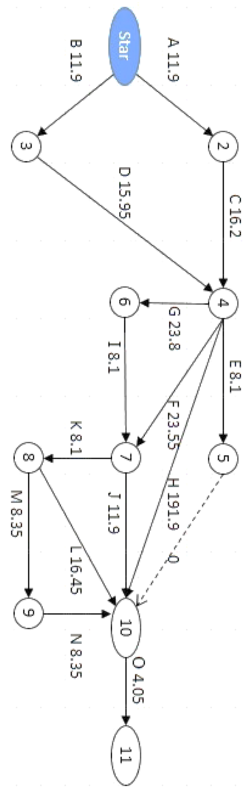

Figure 2. Network Project with crisp number\Source [6].

The branch and bound algorithm for solving the problem are presented:

One of the most important characteristics of the branch and bound algorithm is that of distinguishing the nearest link in the network. Here it must be noted that the nearest link is (Start-3-4-5-10-11) and its total time (40). As shown in figure 3.

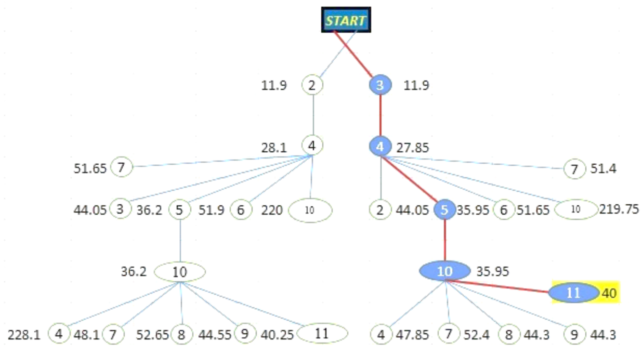

Figure 3. A tree search for finding the nearest link of *node* (11) without improvement for network.

THE IMPROVEMENTS

Each improvement in the network was represented in figure 2 that shows a new variation of the base algorithm, and an increase in its solving efficiency concerning the finding of the nearest link to *node* (11). It would be of great interest to notice the hybrid- resulting effect of such improvements, as follows:

1- Build a link between nodes (3) and (6);
2- Build a link between nodes (3) and (8);
3- Build a link between nodes (3) and (7);
4- Build a link between nodes (3) and (6, 7, 8);

METHODOLOGY AND PROBLEM STATEMENT

An exact approach to solve any optimization network problem is to implicitly enumerate all solutions. Branch-and-bound (B&B) is a well-known technique largely applied to many similar problems. It can be tailored in order to get efficient solutions quickly. We shall now describe how nearest link are computed and how the branching variable selections are made. [5]

1. Build a link between nodes (3) and (6)

Depending on figure 2 the node (3) will be linked to the node (6) in order to find the best and nearest link to the *node* (11), which is shown in figure 4:

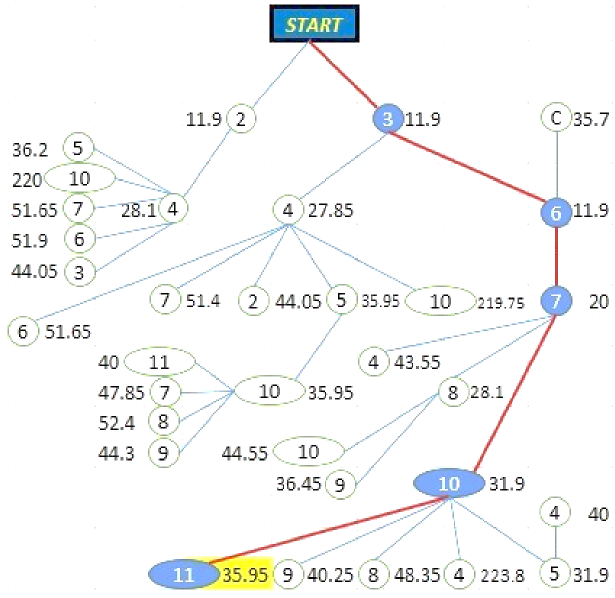

Figure 4. A tree search to find the nearest link to *node* (11) by joined node (3) in node (6)

2. Build a link between nodes (3) and (8)

Depending on figure 2, the node (3) will be linked to the node (8) in order to find the best and nearest link to the *node* (11), which is shown in figure 5:

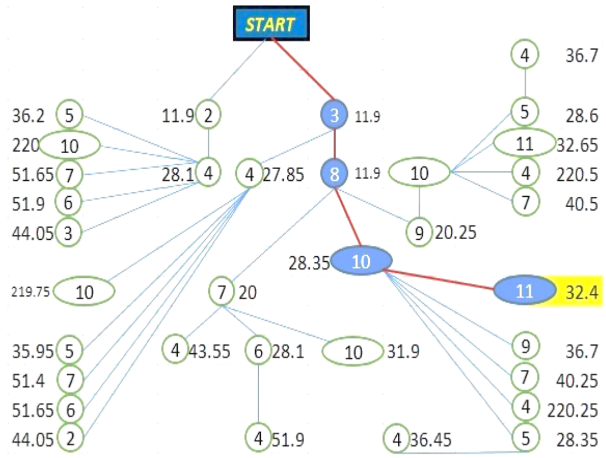

Figure 5. A tree search to find the nearest link to *node* (11) by joined node (3) in node (8).

3. Build a link between nodes (3) and (7)

Depending on figure 2, the node (3) will be linked to the node (7) in order to find the best and nearest link to the *node* (11), which is shown in figure 6:

ISSN: 1814-635X (print), ISSN:2521-3520 (online) Volume 31, Issue 4, 2020 DOI: http://doi.org/10.23851/mjs.v31i4.923

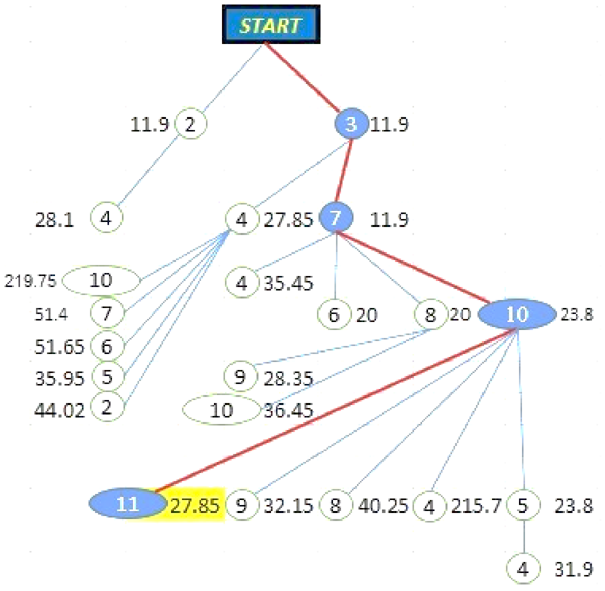

Figure 6. A tree search to find the nearest link to *node* (11) by joined node (3) in node (7).

4. Build a link between nodes (3) and (6, 7, 8)

Depending on figure 2, the node (3) will be linked to the node (6, 7, 8) in order to find the best and nearest link to the *node* (11), which is shown in figure 7:

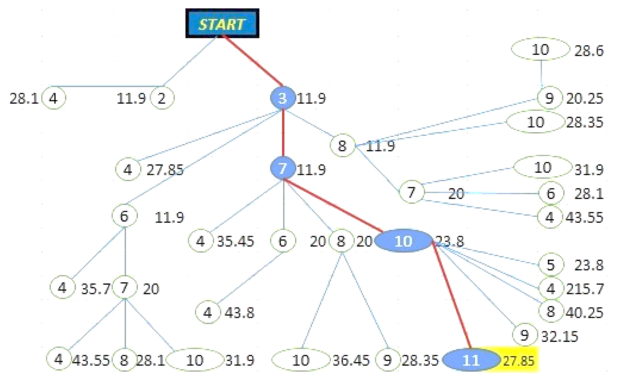

Figure 7. A tree search to find the nearest link to *node* (11) by joined node (3) in node (6, 7, 8).

IMPROVEMENT EVALUATION AND RESULTS DISCUSSION

In this section, we evaluate the first three cases separately, and then evaluate the last two cases.

The first case gave the nearest link represented by the nodes (start-3-4-5-10-11) with a total value of 40, which represent activities (B, D, E, Dum, O), which is a pessimistic for the company. The second case gave the nearest link represented by the nodes (start-3-6-7-10-11) with a total value of 35.95, which represent activities (B, Dum, I, J, O).

The third case gave the nearest link represented by the nodes (start-3-8-10-11) with a total value of 32.4, which represents activities (B, Dum, L, O), for each of first three cases, We notice that there is a change in the objective function, and this is a good indicator for decision makers, the branch and bound algorithm may have the ability to improve the solutions found by the first three cases, the total value in first case is 40, while it drops to 32.4 when using the branch and bound algorithm at the third case.

The fourth and fifth cases gave the same of value to the objective function about nearest link represented by the nodes (start-3-7-10-11) with a total value of 27.85, which represent activities (B, Dum, J, O), which is an optimistic for the company.

What may confirm the advantage of global optimization is the improvement with respect to branch and bound algorithm, even within the strict time limits imposed by real-time purposes.

The branch and bound algorithm finds proven optimal solutions in (5) cases. In the three open instances, the branch and bound algorithm is able to improve the solutions by the first three cases and enhanced by the fourth and fifth cases.

ACKNOWLEDGEMENTS

The researchers gratefully acknowledge Mustansiriyah University, College of Administration and Economics / Department of Statistics.

REFERENCES

[1] San Segundo, P., Coniglio, S., Furini, F., & Ljubić, I. (2019). A new branch-and-bound algorithm for the maximum edge-weighted clique problem. European Journal of Operational Research, 278(1), 76-90. https://doi.org/10.1016/j.ejor.2019.03.047

[2] Elizabeth, S., & Sujatha, L. (2015). Project scheduling method using triangular intuitionistic fuzzy numbers and triangular fuzzy numbers. Applied Mathematical Sciences, 9(4), 185-198. https://doi.org/10.12988/ams.2015.410852

[3] Haffner, S., Monticelli, A., Garcia, A., & Romero, R. (2001). Specialised branch-and-bound algorithm for transmission network expansion planning. IEE Proceedings-Generation, Transmission and Distribution, 148(5), 482-488. https://doi.org/10.1049/ip-gtd:20010502

[4] Marinescu, R., & Dechter, R. (2009). AND/OR branch-and-bound search for combinatorial optimization in graphical models. Artificial Intelligence, 173(16-17), 1457-1491. https://doi.org/10.1016/j.artint.2009.07.003

[5] Poorzahedy, H., & Rouhani, O. M. (2007). Hybrid meta-heuristic algorithms for solving network design problem. European Journal of Operational Research, 182(2), 578-596. https://doi.org/10.1016/j.ejor.2006.07.038

[6] Chachan, Hanan, A. & Elaibi, Waleed, M. (2020). Using Robust Ranking & Linear Programming Technique for Fuzzy Projects. Italian Journal of Pure and Applied Mathematics, Acceptance Letter No. (45).

[7] San Segundo, P., Coniglio, S., Furini, F., & Ljubić, I. (2019). A new branch-and-bound algorithm for the maximum edge-weighted clique problem. European Journal of Operational Research, 278(1), 76-90. https://doi.org/10.1016/j.ejor.2019.03.047

[8] Shanmugasundari, M., & Ganesan, K. (2014). Project scheduling problems under fuzzy environment: A new solution approach. International Journal of Pure and Applied Mathematics, 95(3), 387-399. https://doi.org/10.12732/ijpam.v95i3.6