How predicates improve solving of supply chain problems using linear programming with binary variables

Vincent Giard

Giard V. How predicates improve solving of supply chain problems using linear programming with binary variables. J Pure Appl Maths. 2024; 8(6):1-2.

ABSTRACT

This paper, which focuses on supply chain management issues, demonstrates how predicates can reduce the existence domain of binary decision variables used in deterministic linear problems whose formulation is based on an AML-based approach (Algebraic Modelling Language). The AML-based software employs a generic problem description that is closely aligned with its mathematical formulation and a dataset to create an instance of the problem to be solved. This generic description, independent of the data, can mobilize predicates that utilize parameters from the dataset in the instance creation, thereby reducing the existence domain of binary decision variables.

Key Words: *Algebraic modelling language; Binary variable; Predicate; Solving performance, Assignment*

INTRODUCTION

This paper, which focuses on supply chain management problems, demonstrates how predicates can reduce the existence domain of binary decision variables used in deterministic linear problems, with a particular focus on the assignment problem, when its formulation is based on an AML-based (Algebraic Modelling Language) approach [1-5].

The AML-based software (e.g., Xpress or GAMS) combines a generic problem description, which is closely aligned with its mathematical formulation, and a dataset to create an instance of the problem to be solved. This generic description, independent of the data, can mobilize predicates that utilize parameters from the dataset in the instance creation, thus reducing the existence domain of binary decision variables. This drastic reduction is illustrated in Azzamouri [6-9].

Two important points need to be made on the scope of the analysis presented. First, using predicates to reduce a priori the solution search space may also be possible when the decision variable is integer or continuous (to be discussed at the end of this paper). Secondly, this discussion focuses on decision variables and does not cover auxiliary variables (calculated from decision variables) used to determine the state of specific resources (e.g., the use of a shared resource over time).

Binary decision variables in linear optimization problems can be classified into two categories:

1. In the first case, the binary variables are used to describe choices concerning one or more decisions to be made by making their realization conditional on logical conditions; this is exemplified by the classic problem of selecting investments within budgetary constraints or of scheduling projects without considering more than one resource shared by the activities.
2. In the second case, the binary variable is used to solve assignment problems aimed at linking objects belonging to different classes of objects (in the broad sense) corresponding to resources (people, machines, sheds, factories, berths, etc.), as well as time. The use of predicates is of interest only for this second category.

Case 1

In the first case, a binary variable contains only one subscript corresponding to a simple decision number, and the model created generally includes a set of existence conditions linking these choices (incompatibility, conditional realization, inclusion, etc.) via relations linking these variables. The scientific and commercial literature lists these logical conditions.

Some scheduling problems aim to determine the sequence of K products passing through a production system (flow-shop, assembly line), which leads to using a binary variable with two indices (of equal cardinality), the first for the product and the second for its rank. If these binary variables have no additional subscripts (e.g., linked with resources), the scheduling problem falls into the first case.

Université Paris-Dauphine, PSL Research University, Paris, France

Correspondence: Vincent Giard, Université Paris-Dauphine, PSL Research University, Paris, France, E-mail: vincent.giard@dauphine.psl.eu

Received: 21 Oct, 2024, Manuscript No. puljpam-24-7237, Editor Assigned: 23 Oct, 2024, PreQC No. puljpam-24-7237 (PQ), Reviewed: 27 Oct, 2024, QC No. puljpam-24-7237 (Q), Revised: 31 Oct, 2024, Manuscript No. puljpam-24-7237 (R), Published: 30 Nov, 2024, DOI:-10.37532/2752-8081.24.8(6).01-02

Case 2

In the second case, a binary variable has as many subscripts as there are sets of objects in the assignment problem, for example, "vessel berth period t", in a simplified port management problem. In this example, there are possible binary variables for a problem with V vessels, B berths and T periods. Several attributes can characterize each element in a set. For example, the element "vessel v" may have 3 attributes: its length, its loaded draught and its earliest arrival time; and the element "berth b" may have two attributes: its length and its depth. One of these sets is privileged in the perspective retained in the assignment problem (in our example, it is the vessel) and is referred to here as entities.

Finally, it should be noted that in some scheduling problems involving a resource k (among K possible resources), two indices i and j may refer to two objects from the same class of objects to be processed successively on the same resource (e.g. order) and not from two different classes: the binary variable worths 1 if these two entities i and j are processed successively by the same resource k.

Constraints linked to the attributes of the objects of classes represented by the subscripts of the binary variable.

A distinction must be made between cases where displacement is not explicitly involved and those where it must be considered.

Analysis of the displacement ignored case in the assignment problem.

After presenting the classical solution based on the introduction of explicit constraints using an example, we will examine the solution based on predicates. We then present further examples of interesting predicate use for binary decision variables, in which time is mobilized as a subscript.

Classical formulation using constraints

Some constraints concern the existence of a binary variable that must consider forbidden combinations between specific values of the attributes of each set mobilized in the multidimensional binary variable. For example: *i*) vessel *v* can only dock at berth *b* if its loaded draught d_v is less than the depth D_p of that berth and if its length l_v does not exceed the berth length L_p ; *ii*) vessel *v* cannot dock before its arrival A_v in port. Assuming that all vessels must dock in the port $\left[\sum_{bt} x_{vbt}=1, \forall v\right]$, these constraints are traditionally addressed by the following three relations: $\left[\sum_t \mathrm{d}_v \cdot x_{vbt} \le \mathrm{D}_p, \forall v, b\right]$; $\left[\sum_t \mathrm{l}_v \cdot x_{vbt} \le \mathrm{L}_p, \forall v, b\right]$; $\left[\sum_b t \cdot x_{vbt} \ge \mathrm{A}_v, \forall v, t\right]$.

Introducing these constraints, whose purpose is to progressively restrict the search space in solving the problem, is unnecessary with predicates when using AML-based software, simplifying the problem's numerical resolution.

Solution based on the use of predicates.

CONCLUSION

Using predicates in optimization problems with binary decision variables (case 2) handled by AML-based software can significantly reduce the size of the problem posed (as attested in the reference articles) and avoid the need to create specific solution algorithms. Furthermore, by facilitating the resolution of complex problems, researchers can take into account more realistically a certain number of constraints of the problem posed in the field, (such as the simultaneous use of a fine temporal granularity to model system operation and a variable decision step which can be increased for remote decisions), often ignored for the impossibility of finding a solution numerically. It should be noted that the principles described in this note for SC management problems can be easily adapted to other categories of optimization problems using binary variables in software based on the AML approach.

REFERENCES

1. Cook S. The importance of the P versus NP question. J. ACM(JACM). 2003 1;50(1):27-9.
2. Rabin MO, Scott D. Finite automata and their decision problems. IBM J Res Dev. 1959 3(2):114-25.
3. Berry, Gerard, Ravi Sethi. From regular expressions to deterministic automata. Theoretical Computer Science 48 (1986): 117-126.
4. Schneider, Klaus, Jimmy Shabolt, et al. Verification of reactive systems: formal methods and algorithms. Heidelb Springer 2004.
5. Syzdykov, Mirzakhmet. Functional hypothesis of complexity classes. Adv. Technol. and Comput. Sci. 3 (2022): 4-9.
6. Syzdykov M. Membership Problem in Non-deterministic Finite Automata for Extended Regular Expressions in Linear Polynomial Time. Adv Technol Comput Sci. 2021 20(4):14-7.
7. Brzozowski JA. Derivatives of regular expressions. J. ACM (JACM). 1964 1 11(4):481-94.
8. Antimirov V. Partial derivatives of regular expressions and finite automaton constructions. Theor. Comput. Sci. 1996 11.155(2):291-319.
9. Bellman, Richard. The theory of dynamic programming. Bull. Am. Math. Soc. 60.6 (1954): 503-515.