The Distance Constraint on Sequence Variables*

Margaux Schmied
Université Côte d'Azur, France

Augustin Delecluse
KU Leuven, Belgium

Jean-Charles Régin
Université Côte d'Azur, France

Pierre Schaus
UCLouvain, Belgium

Abstract

Insertion sequence variables have recently been introduced as a computational domain for modeling routing and sequencing problems in constraint programming. This paper investigates filtering for the (minimum) distance constraint over insertion sequence variables. This global constraint links a sequence to a distance variable based on a given distance matrix. So far, only a simple filtering algorithm has been proposed, which considers partial paths but ignores mandatory nodes. Our contribution introduces stronger lower bounds by integrating mandatory nodes. These bounds further enable the derivation of additional filtering rules for node insertions.

2012 ACM Subject Classification Theory of computation → Constraint and logic programming

Keywords and phrases Constraint programming, sequence variable, global constraint, cost-based constraint, distance constraint, lower bound

Digital Object Identifier 10.4230/LIPIcs.CVIT.2016.23

Acknowledgements This work has been supported by the French government, through the 3IA Côte d'Azur Investments in the Future project managed by the National Research Agency (ANR) with the reference number ANR-19-P3IA-0002.

1 Introduction

Routing and scheduling problems are central in constraint programming applications such as vehicle routing [3, 4], scheduling [5], and the Traveling Salesman Problem (TSP)[6]. Sequence variables [10] provide a flexible framework to model such problems by incrementally constructing solutions through node insertions into a partial sequence while enforcing ordering constraints.

Let V be a finite set of nodes. A sequence $\vec{S}$ is an ordered list without repetition, starting at α and ending at ω and $\vec{s}$ a subsequence. A sequence variable domain is defined by a partial sequence $\vec{s}$, a set of required nodes R, a set of excluded nodes X, and placement constraints. Domain update consists in inserting nodes from $I = V \setminus (\vec{s} \cup X)$ into $\vec{s}$. An insertion (v_i, v_j, v_k) inserts v_j between consecutive nodes v_i and v_k.

Figure 1 shows a sequence domain before and after an insertion.

Constraint propagation over sequence variables is a key component of this framework [1], in particular for cost-based constraints such as the distance constraint. This global constraint links a sequence $\vec{S}$ to $Dist$, its total travel cost: $Dist = \sum_{v_i \xrightarrow{\vec{s}} v_j} d_{i,j}$.

Propagating the distance constraint is challenging due to the combinatorial nature of insertion decisions. Existing approaches rely on relaxed propagation [2], considering insertions independently. More precisely, propagation only reasons on lower bounds computed on the current partial sequence:

* This paper is an extended abstract of a paper accepted at the main conference [8].

42nd Conference on Very Important Topics (CVIT 2016).
Editors: John Q. Open and Joan R. Access; Article No. 23; pp. 23:1–23:3
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

$L_O(\vec{s}) = \sum_{v_i \xrightarrow{\vec{s}} v_j} d_{i,j}$, while ignoring the later cost of insertion of required insertable nodes that must be inserted later. For each insertion (v_i, v_j, v_k), the associated detour cost is: $\Delta_{i,j,k} = d_{i,j} + d_{j,k} - d_{i,k}$. Insertions exceeding the upper bound are removed: $L_O(\vec{s}) + \Delta_{i,j,k} > \lceil Dist \rceil$.

In this paper, we strengthen this reasoning by introducing admissible lower bounds that explicitly account for required insertable nodes. These bounds are derived from relaxations of the insertion process and enable additional filtering rules for infeasible insertions.

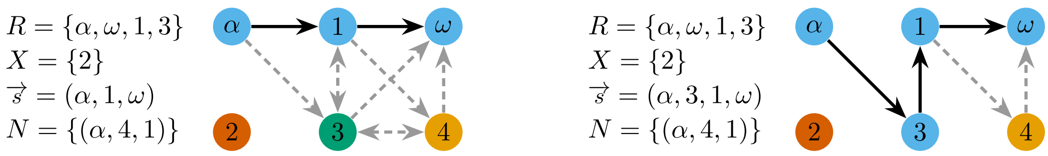

Figure 1 Sequence domain representation. The partial sequence $\vec{s}$ is represented with solid bold lines, while the remaining edges from the graph are shown with dashed gray lines. The right part shows the domain from the left part after inserting the required node 3 into $\vec{s}$.

2 Contribution

Our work introduces three relaxations of the insertion process, leading to stronger lower bounds and associated filtering rules.

The first relaxation considers that every required node must have at least one predecessor and one successor in the final sequence. Ignoring connectivity constraints, we assign each required node independently to its cheapest predecessor (or successor). The resulting bound sums the minimum incoming (or outgoing) arc costs: $L_{MIO}(\vec{s}) = \max\left(\sum_{v_j \in R} \min_{v_i \in R} d_{i,j}, \sum_{v_j \in R} \min_{v_k \in R} d_{j,k}\right)$.

This bound can also be exploited for filtering. When evaluating an insertion (v_i, v_j, v_k), the minimum predecessor or successor contribution previously associated with v_j is removed and replaced by the actual detour induced by the insertion.

The second relaxation reasons directly on insertions. Each required insertable node is forced to contribute independently through its minimum insertion detour, while interactions between insertions are ignored. This gives the following lower bound: $L_{MD}(\vec{s}) = L_O(\vec{s}) + \sum_{v_j \in R \cap I} \min_{v_i, v_k \in R} \Delta_{i,j,k}$.

Similarly to the previous relaxation, filtering is achieved by replacing the minimum detour associated with a node by the actual detour induced by the considered insertion.

The third relaxation approximates the global predecessor and successor structure of the sequence without enforcing connectivity. Each node is constrained to have exactly one predecessor and one successor, while disconnected components are allowed. The resulting lower bound is obtained through a minimum cost maximum flow (`MinCostMaxFlow`), similarly to approaches used for the cardinality constraint with costs [7, 9].

3 Conclusion

We introduced three admissible lower bounds for the distance constraint on sequence variables, derived from relaxations of the insertion process. Experiments [8] show that the detour-based bound is particularly effective in reducing the search space and improving performance, highlighting the benefit of reasoning on future insertions. These results demonstrate the potential of insertion-based relaxations to strengthen propagation in sequence variables.

References

1. Augustin Delecluse, Pierre Schaus, and Pascal Van Hentenryck. Sequence variables for routing problems. In *28th International Conference on Principles and Practice of Constraint Programming (CP 2022)*, pages 19–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2022.
2. Augustin Delecluse, Pierre Schaus, and Pascal Van Hentenryck. Sequence variables: A constraint programming computational domain for routing and sequencing. Manuscript in preparation, 2025.
3. Frédéric Didier, Laurent Perron, Sarah Mohajeri, Steven Agostino Gay, Thibaut Cuvelier, and Vincent Furnon. Or-tools' vehicle routing solver: a generic constraint-programming solver with heuristic search for routing problems. 2023.
4. Mustafa Küçük and Seyda Topaloglu Yildiz. Constraint programming-based solution approaches for three-dimensional loading capacitated vehicle routing problems. *Computers & Industrial Engineering*, 171:108505, 2022.
5. Roberto Montemanni and Mauro Dell'Amico. Constraint programming models for the parallel drone scheduling vehicle routing problem. *EURO Journal on Computational Optimization*, 11:100078, 2023.
6. Gilles Pesant, Michel Gendreau, Jean-Yves Potvin, and Jean-Marc Rousseau. On the flexibility of constraint programming models: From single to multiple time windows for the traveling salesman problem. *European Journal of Operational Research*, 117(2):253–263, 1999. URL: `https://www.sciencedirect.com/science/article/pii/S0377221798002483`, `doi:10.1016/S0377-2217(98)00248-3`.
7. Jean-Charles Régin. Cost-based arc consistency for global cardinality constraints. *Constraints*, 7(3/4):387–405, jul 2002. `doi:10.1023/A:1020506526052`.
8. Margaux Schmied, Augustin Delecluse, Pierre Schaus, and Jean-Charles Régin. The distance constraint on sequence variables. In *International conference on principles and practice of constraint programming*, 2026.
9. Margaux Schmied and Jean-Charles Régin. Efficient implementation of the global cardinality constraint with costs. In *30th International Conference on Principles and Practice of Constraint Programming (CP 2024)*, pages 27–1. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2024.
10. Charles Thomas, Roger Kameugne, and Pierre Schaus. Insertion sequence variables for hybrid routing and scheduling problems. In *International Conference on Integration of Constraint Programming, Artificial Intelligence, and Operations Research*, pages 457–474. Springer, 2020.