Extended Abstract: Acquiring Maps of Interrelated Conjectures on Sharp Bounds

Nicolas Beldiceanu
IMT Atlantique, LS2N (TASC), Nantes, France
Jovial Cheukam-Ngouonou[1]
IMT Atlantique, LS2N (TASC), Nantes, France, and Université Laval, Québec, Canada
Rémi Douence
IMT Atlantique, LS2N, Inria, (Gallinette), Nantes, France
Ramiz Gindullin
IMT Atlantique, LS2N (TASC), Nantes, France
Claude-Guy Quimper
Université Laval, Québec, Canada

1 Introduction

Research on conjectures making systems in the context of discrete mathematics is a topic that goes back to the late 1950s [10]. Within CP, some initial research on the generation of implied constraints was done [5] and the most recent work focuses on model and constraint acquisition [9] rather than on conjecture making. Within OR, Hansen's AutoGraphiX system [7] focuses on finding unrelated bounds using Variable Neighbourhood Search.

Four reasons motivate our work: (i) to highlight that CP can contribute to the automatic discovery of conjectures, (ii) to systematically search sharp bounds on characteristics of objects that show up in combinatorial problems, (iii) to stress the need to develop strongly interrelated knowledge, (iv) by the fact that bounds are an essential feature of branch-and-bound methods in optimisation but also a weakness of CP [6, 8]: the development of sharp bounds that consider several interrelated characteristics is still a manual process [2, 4]. Our approach is unique among all works for conjectures generation, as the result is not a set, but rather a *graph of conjectures*, linked by projection operators. Our contributions are:

- We introduce the concept of *map of sharp bounds* as a set of interrelated conjectures providing sharp lower and upper-bounds wrt the characteristic of a combinatorial object.
- For each conjecture on a sharp bound, the map gives some *extremal characteristics* i.e., the characteristic values common to all combinatorial objects achieving the bound.
- We demonstrate the usefulness of CP for acquiring such maps over digraphs.

The significance of maps is twofold. Beyond sharp bounds, a map brings together the relations between several sharp bounds and the structure of combinatorial objects reaching each bound under the same edifice.

We introduce the concept of a *map* that presents a set of conjectures for sharp bounds and their logical relations.

2 Conjectures map as a symbolic piece of knowledge

We introduce the concept of a *map of conjectures* as a way to reveal the links between a set of conjectures related to sharp bounds for a characteristic of a combinatorial object. Our goal is to describe conjectures on sharp bounds of characteristics of a combinatorial object, e.g. a digraph, a tree, and to organise these conjectures into a single structure, a *map of sharp*

[1] Participating student to the doctoral program

The Twenty-Seventh Conference on Principles and Practice of Constraint Programming (CP 2022).
Editors: Christine Solnon; Article No. 1; pp. 1:1–1:4
OASICS OpenAccess Series in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

bounds, which (i) systematically interconnects these conjectures, and which (ii) describes the structure of the combinatorial objects for which the bounds are reached.

▶ **Definition 1.** *Given a finite set of input characteristics $\mathcal{P}$ and an output characteristic $\mathrm{o} \notin \mathcal{P}$, a* map of sharp upper bounds $\mathcal{M}_{\mathcal{P}}^{\mathrm{o}\leq}$ *is defined as a digraph where:*

- *Each node of the map is associated with a subset $P \subseteq \mathcal{P}$ of input characteristics and corresponds to a* maximum conjecture *of the form $\mathrm{o} \leq f(P)$. This inequality is tight, i.e. there exist values that can be given to the parameters P in order to reach the equality.*
  In addition, a node contains maximality conjectures*, at most one per characteristic q in the complement of P wrt $\mathcal{P}$, represented by the symbolic equality $q = g_q(P)$, where g_q is a function defined over realisable parameters values of P and called a* maximum characterisation*, and expressing the following property: for any combination of parameters P reaching the maximum $f(P)$, the characteristic q is equal to $g_q(P)$.*
- *Each arc from conjecture $\mathrm{o} \leq f_i(P_i)$ to conjecture $\mathrm{o} \leq f_j(P_j)$ corresponds to a projection from a subset P_i of input characteristics to a subset P_j of input characteristics, by eliminating a characteristic $q_{i,j}$, i.e. $P_j = P_i \setminus \{q_{i,j}\}$. The arc is labelled with an equality $q_{i,j} = g_{q_{i,j}}(P_j)$ where $g_{q_{i,j}}(P_j)$ is the value given to $q_{i,j}$ to reach the equality in the conjecture $\mathrm{o} \leq f_j(P_j)$. The equality $q_{i,j} = g_{q_{i,j}}(P_j)$ is called a* maximality conjecture*.*

In a map, there is one output characteristic that we bound using the other characteristics called input characteristics. The output characteristic is the *bounded characteristic*, while the input characteristics are the *bounding characteristics*. The maximum conjecture provides a bound on the output characteristic wrt the characteristics in P. The maximality conjectures indicate the values taken by the characteristics not in P when the bound is reached.

▶ **Example 2.** Fig. 1 illustrates the map concept. As an example of combinatorial objects, we use in this paper digraphs with these characteristics: the number v of vertices, the number a of arcs, the number c (resp. s) of connected components (resp. strongly connected components), the number $\underline{c}$ of vertices of the smallest connected component.

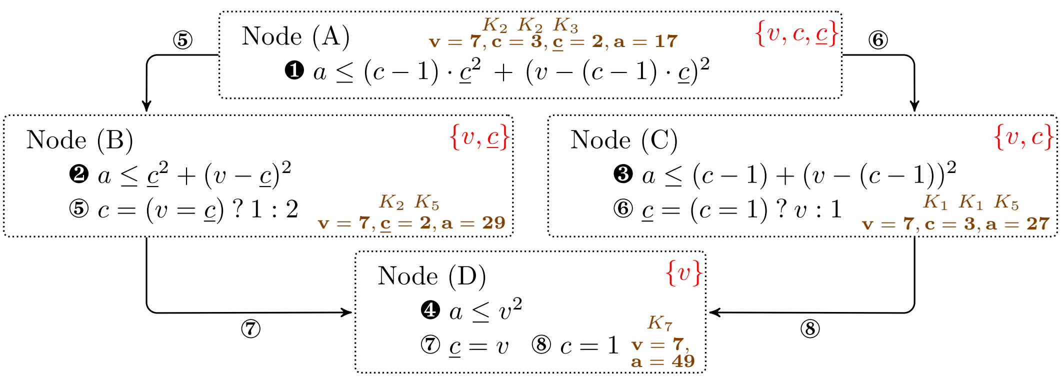

Figure 1 Map $\mathcal{M}^{a\leq}_{\{v,c,\underline{c}\}}$ with the sharp upper-bounds ❶, ❷, ❸, ❹ for the number of arcs in a digraph; each node presents an example in brown: given a value for the characteristics attached to the node, a graph reaching the maximum is described, as a union of cliques K_i, with i vertices, e.g. in node (B), given the assignments $v = 7$ and $\underline{c} = 2$, the digraph with 2 cliques K_2, K_5 reaches the maximum 29 for the number a of arcs; *cond* ? $x : y$ denotes x if condition *cond* holds, y otherwise.

2.1 A constraint approach for acquiring symbolic equations

The search for sharp bounds leads to the identification of equations in which the left-hand side is an output or a secondary characteristic, and the right-hand side is a formula involving input and secondary characteristics. Rather than applying a bottom-up approach that generates formulae of increasing complexity, we adopt the following strategy. As we aim at finding simple formulae, we use three complementary classes of formulae that turned out to appear concomitantly in a map: (1) Boolean formulae involving k arithmetic conditions linked by a single commutative logical operator or by a sum, (2) simple conditional formulae, and (3) formulae over polynomials that can share common sub-expressions. A first attempt to use only polynomials without common sub-expressions missed some formulae and quite often provided too complicated formulae. Based on the metadata, the CP approach restrict the space of possible formulae like those in the map of Fig. 1, generate parameterised candidate formulae and finally fix the values of their parameters using data.

Comparing the conjectures founds with proved bounds of the constraint catalogue [1], the Bound Seeker retrieves 66.66% of the bounds of the constraint catalogue, even if the resulting formulae have sometimes a different form.

3 Conclusion

We introduce a structure that connects a set of sharp bounds. Based on this structure, we propose a constructive approach to acquire a set of interrelated conjectures on sharp bounds. This work opens a new application domain for CP for automated conjectures-making systems. It creates a new line of research to those already reported in a recent survey on machine learning for combinatorial optimisation [3].

References

1 Nicolas Beldiceanu, Mats Carlsson, and Jean-Xavier Rampon. Global Constraint Catalog, 2nd Edition (revision a). Technical Report T2012-03, Swedish Institute of Computer Science, 2012. Available at http://ri.diva-portal.org/smash/get/diva2:1043063/FULLTEXT01.pdf.

2 Nicolas Beldiceanu, Mats Carlsson, Jean-Xavier Rampon, and Charlotte Truchet. Graph invariants as necessary conditions for global constraints. In Peter van Beek, editor, *Principles and Practice of Constraint Programming - CP 2005, 11th International Conference, CP 2005, Sitges, Spain, October 1-5, 2005, Proceedings*, volume 3709 of *Lecture Notes in Computer Science*, pages 92–106. Springer, 2005.

3 Yoshua Bengio, Andrea Lodi, and Antoine Prouvost. Machine learning for combinatorial optimization: A methodological tour d'horizon. *Eur. J. Oper. Res.*, 290(2):405–421, 2021.

4 Christian Bessière, Emmanuel Hebrard, George Katsirelos, Zeynep Kızıltan, Émilie Picard-Cantin, Claude-Guy Quimper, and Toby Walsh. The balance constraint family. In Barry O'Sullivan, editor, *Principles and Practice of Constraint Programming - 20th International Conference, CP 2014, Lyon, France, September 8-12, 2014. Proceedings*, volume 8656 of *Lecture Notes in Computer Science*, pages 174–189. Springer, 2014.

5 John William Charnley, Simon Colton, and Ian Miguel. Automatic generation of implied constraints. In Gerhard Brewka, Silvia Coradeschi, Anna Perini, and Paolo Traverso, editors, *ECAI 2006, 17th European Conference on Artificial Intelligence, August 29 - September 1, 2006, Riva del Garda, Italy, Including Prestigious Applications of Intelligent Systems (PAIS 2006), Proceedings*, volume 141 of *Frontiers in Artificial Intelligence and Applications*, pages 73–77. IOS Press, 2006. URL: http://www.booksonline.iospress.nl/Content/View.aspx?piid=1649.

6 Minh Hoàng Hà, Claude-Guy Quimper, and Louis-Martin Rousseau. General bounding mechanism for constraint programs. In Gilles Pesant, editor, *Principles and Practice of*

Constraint Programming - 21st International Conference, CP 2015, Cork, Ireland, August 31 - September 4, 2015, Proceedings, volume 9255 of *Lecture Notes in Computer Science*, pages 158–172. Springer, 2015.

7 Pierre Hansen and Gilles Caporossi. Autographix: An automated system for finding conjectures in graph theory. *Electron. Notes Discret. Math.*, 5:158–161, 2000. doi:10.1016/S1571-0653(05)80151-9.

8 Jimmy Ho-Man Lee, Ka Lun Leung, and Yu Wai Shum. Consistency techniques for polytime linear global cost functions in weighted constraint satisfaction. *Constraints*, 19(3):270–308, 2014.

9 Steve Prestwich. Robust constraint acquisition by sequential analysis. In Giuseppe De Giacomo, Alejandro Catalá, Bistra Dilkina, Michela Milano, Senén Barro, Alberto Bugarín, and Jérôme Lang, editors, *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 355–362. IOS Press, 2020. doi:10.3233/FAIA200113.

10 Hao Wang. Toward mechanical mathematics. In Jörg Siekmann and Graham Wrightson, editors, *Automation of Reasoning: Classical Papers on Computational Logic 1957–1966*, pages 244–264. Springer-Verlag, Berlin, 1983.