The use of restrictions in Logic Programming: Puzzle Akkoy

Filipa Ramos and Ines Santos

Faculdade de Engenharia da Universidade do Porto,
Rua Dr. Roberto Frias, 4200 - 465, Porto, Portugal
{up201305378,up201303501}@fe.up.pt
FEUP-PLOG,Turma3MIEIC06,GrupoAkkoy_2
https://sigarra.up.pt/feup/pt/web_page.inicial

Abstract. The present report serves the purpose of explaning the process of finding a solution for akkoy puzzles using programming with restrictions. It also refers the implementation of restrictions in order to generate random puzzles. The main objective is to deepen the knowledge of *PROLOG*, specially the clpfd library. The project was developed for the curricular unit of Logic Programming. The results will be evaluated in order to realize the efficiency of the found solution.

Keywords: restrictions programming logic prolog efficiency

1 Introduction

This project was developed for the curricular unit of Logic Programming in order to deepen the knowledge of the clpfd prolog library. Between the many objectives accounted for this project, the following can be highlighted: examining the results of the use of restrictions whilst programming, understanding the logic of rule-based languages, realizing the advantages of logic in programming. The analized puzzle has many restrictions which made it hard to find a solution which incorporated all the restrictions. Besides this, the efficiency of the solution can not be hightened as well as in other puzzles since the difficulty level is very high. The implemented solution uses the following approach: count the number of black squares in columns and white squares in lines and compare it with the numbers on the restrictions list. The returned board is a list of variables. Each number one represents a black square and each number two represents a white square. The article is structured in order to make it easier to understand the solution. Firstly, the problem is described. Secondly, the solution and its visualization is explained. Finally, the results are analized and the conclusions are made.

2 Problem Description

3 Approach

3.1 Decision Variables

3.2 Constraints

3.3 Search Strategy

4 Solution Presentation

5 Results

The numbers accorded to lemmas, propositions, and theorems, etc. should appear in consecutive order, starting with Lemma 1, and not, for example, with Lemma 11.

6 Conclusions and Future Work

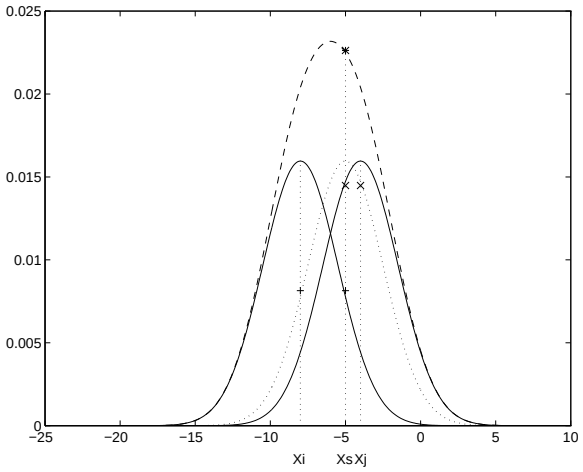

Fig. 1. One kernel at x_s (*dotted kernel*) or two kernels at x_i and x_j (*left and right*) lead to the same summed estimate at x_s. This shows a figure consisting of different types of lines. Elements of the figure described in the caption should be set in italics, in parentheses, as shown in this sample caption.

7 References

References

1. Smith, T.F., Waterman, M.S.: Identification of Common Molecular Subsequences. J. Mol. Biol. 147, 195–197 (1981)
2. May, P., Ehrlich, H.C., Steinke, T.: ZIB Structure Prediction Pipeline: Composing a Complex Biological Workflow through Web Services. In: Nagel, W.E., Walter, W.V., Lehner, W. (eds.) Euro-Par 2006. LNCS, vol. 4128, pp. 1148–1158. Springer, Heidelberg (2006)
3. Foster, I., Kesselman, C.: The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann, San Francisco (1999)
4. Czajkowski, K., Fitzgerald, S., Foster, I., Kesselman, C.: Grid Information Services for Distributed Resource Sharing. In: 10th IEEE International Symposium on High Performance Distributed Computing, pp. 181–184. IEEE Press, New York (2001)
5. Foster, I., Kesselman, C., Nick, J., Tuecke, S.: The Physiology of the Grid: an Open Grid Services Architecture for Distributed Systems Integration. Technical report, Global Grid Forum (2002)
6. National Center for Biotechnology Information, `http://www.ncbi.nlm.nih.gov`

8 Annex

$$\psi(u) = \int_o^T \left[\frac{1}{2}\left(\Lambda_o^{-1}u, u\right) + N^*(-u)\right] dt\ . \tag{1}$$