Adaptive Combinatorial Search

PhD student: Alejandro Arbelaéz
Supervisors: Youssef Hamadi and Marc Shoenauer

MICROSOFT RESEARCH INRIA JOINT CENTRE

UNIVERSITÉ PARIS-SUD 11

Abstract: The main goal, is to set automatic tuning methods for CSP algorithms, allowing e-scientists who have little knowledge of the search technique itself to nevertheless solve their optimization problem without the need for some optimization engineer. We will address both off-line and on-line tuning issues, at problem level as well as instance level. These techniques will be tested on several computational biological problems.

Introduction

A Constraint Satisfaction Problem (CSP) is a triple (X, D, C) where:

- X represents a set of n variables.
- D represents the set of variables domains, i.e possible values for the variables.
- C represents a finite set of constraints.

Solving a CSP involves the finding of consistent values for all variables or the demonstration that no solution can be found.

Search

A complete search procedure can be used to solve a CSP. To be efficient, it has to be combined with a good search-strategy which involves the successive selection of variable/value pairs:

- Choose a variable at each node in the search tree (Variable Selection)
- Choose a value for the selected variable (Value Selection)

Problem	domWdeg	domFD
all-interval	32. 35	25. 37
partition	2. 59	1.31
bibd	4. 56	0.04

Table 1: average runtime (s) per benchmark-family

- When an inconsistency is found, a backtrack procedure is called.

Ongoing work

Usually, search-strategies are defined for a class of problems. Our goal is to adapt the strategy to the problem at-hand. In this way, our research directions are:

Variable Selection:

Heuristically discover weak dependencies between variables, to guide the search procedure, domFD.

Table 1 shows a comparison against domWdeg, one of the best dynamic variable ordering heuristics.

Automatic Parameter Configuration:

Applying a machine learning technique called linear regression in order to predict the best search-strategy.

- Identify a suitable set of features, in the context of CSP's.
- Applying a Ridge regression in order to learn a function f(x) which is going a be a run-time predictor.
- Predict the best parameter configuration for an unseen instance based on the learnt function.

General Discussion

- domFD has shown a search reduction by several orders of magnitude.
- Combining domFD with domWdeg.
- Setting the best parameter configuration per instance we will improve the over all performance for a class of problems.
- Adapt our search-strategy during the resolution process.
- Our approach should be general enough to be applied to a large number of parameters.