`PrefixCCFWC`: technical report

Victor Lecomte

September 3, 2015

Abstract

In scheduling, it may be useful to specify cardinality constraints on certain prefixes of an array of variables. For example, if only one item can be produced each day, and you have to deliver 3 units of product A after 7 days, you will want to impose that there be at least 3 occurences of A among the production variables for the first 7 days.

This constraint allows you to apply many such constraints on several values without the overhead of creating a GCC for each of them, and with some additional pruning.

Contents

1 Problem statement

We are given an array of integer variables and a number of constraints of the form:

> There should be at least/most b occurrences of value v among the i first variables.

In scheduling this will mostly be lower bounds coming from quantities to be produced at a certain date, but there could also be upper bounds coming from storage limitations.

For example let's consider four variables with values either `A` or `B`, and the following constraints:

- there should be at least one `B` in the first two variables ($b = 1$, $v =$ `B`, $i = 2$);
- there should be at most two `A`s in all the variables ($b = 2$, $v =$ `A`, $i = 4$).

Then the following results would be valid:

- `A B A B`
- `B B B B`

While the following results would be invalid:

- `A A B B` (no `B` in the first two variables)
- `B A A A` (too many `A`s)

2 The algorithm

TODO

2.1 Bound deduction and filtering

TODO

2.2 The concept of critical values

TODO

2.3 Merging and pruning

TODO

2.4 Complexity

TODO

3 Conclusion and use cases

TODO