Assignment 2 **Kim Hammar**
ID2204 **Mallu**
Constraint Programming Due Date: 30 April 2017

Simple linear inequality

Propagator Definition

Propagator $p_{lin} \in c \equiv a \cdot x + b \cdot y \leq c$ where x, y are variables and a, b, c are integer constants

$$p_{lin}(s) = \begin{cases} x \mapsto \{n \in s(x) | n \leq \frac{c-(b \cdot \min s(y))}{a}\} \\ y \mapsto \{n \in s(y) | n \leq \frac{c-(a \cdot \min s(x))}{b}\} \end{cases}$$

Idempotence

Theorem 0.1. *The propagator p_{lin} is idempotent*

Proof sketch:

Proof. Assume by contradiction that p_{lin} is not idempotent, then there exists two stores $p_{lin}(s) = s'$ and $p_{lin}(s') = s''$ such that $s' \neq s''$. If s' is failed then s'' must also be failed by the definition of the propagator. From now on we therefore consider the case where s' is not failed.

p_{lin} is by definition contracting which means that if $s' \neq s''$ the following must hold: $s'' < s'$, which implies that $s''(x) \subset s'(x) \vee s''(y) \subset s'(y)$.

By definition of p_{lin}, $\forall n \in s'(x), n \leq \frac{c-(b \cdot \min s(y))}{a}$ and $\forall n \in s'(y), n \leq \frac{c-(a \cdot \min s(x))}{b}$. From this it follows that for it to be possible that $s'' < s'$ it must be that $\min s'(x)$ is less than $\min s(x)$ or $\min s'(y)$ is less than $s(y)$. This is a contradiction of the contracting-property of p_{lin} since the minimum value of x or y can only be decreased from store s to s' if a value $v \leq \min s(x)$ is added to the domain of x or a value $w \leq \min s(y)$ is added to the domain of y. □

Subsumption

Detect subsumption:

$$\begin{gathered} p_{lin}(s) = \textbf{let } s' = p_{lin}(s) \\ \textbf{if } a \cdot \max s(x) + b \cdot \max s(y) \leq c \\ \textbf{then } \langle subsumed, s' \rangle \textbf{ else } \langle fix, s' \rangle \end{gathered}$$

Changing Propagation Order

No, it is not true. Assuming a propagator exhibits the properties described and proved in the course notes [1], we know that when executing a set of propagators on a store s until all propagators are at fixpoint, the fixpoint will be the weakest simultaneous fixpoint, no matter in which order the propagators were applied. I.e the order of propagation does not matter when running until simultaneous fixpoint. However, this does not imply that $p_1(p_2(s)) \neq p_2(p_1(s))$ if the propagators are not at fixpoint.

Theorem 0.2.

$$p_1(p_2(s)) \neq p_2(p_1(s)) \textit{ for all propagators/stores}$$

Proof. By construction of counterexample.

Let s_1 be a constraint store and p_1, p_2 be two correct propagators according to the definition in [1], i.e monotonic, contracting and solution preserving propagators.

$$s_1 = \{x \mapsto \{1, \dots, 10\}, y \mapsto \{1, \dots, 10\}\}$$

p_1 implementing the constraint $x \leq 3$ and p_2 implementing the constraint $x + y \geq 10$.

$$p_1(s) = \begin{cases} x \mapsto \{n \in s(x) | n \leq 3\} \\ y \mapsto s(y) \end{cases}$$

$$p_2(s) = \begin{cases} x \mapsto \{n \in s(x) | n \geq 10 - \max s(y)\} \\ y \mapsto \{n \in s(y) | n \geq 10 - \max s(x)\} \end{cases}$$

$$p_1(s_1) = s_2 = \{x \mapsto \{1, \dots, 3\}, y \mapsto \{1, \dots, 10\}\}$$

$$p_2(s_1) = s_1 = \{x \mapsto \{1, \dots, 10\}, y \mapsto \{1, \dots, 10\}\}$$

$$p_2(s_2) = s_3 = \{x \mapsto \{1, \dots, 3\}, y \mapsto \{7, \dots, 10\}\}$$

$$p_1(p_2(s_1)) = p_1(s_1) = s_2$$

$$p_2(p_1(s_1)) = p_2(s_2) = s_3$$

$$s_2 \neq s_3$$

□

Idempotent Propagators

Yes it is true. This follows from the fact that propagators are contracting and that constraint stores have <u>finite</u> sets of variables and values.

Theorem 0.3.

$$\exists n \in \mathbb{N} \textit{ such that } p^n \textit{ is idempotent for any store } s \textit{ and correct propagator } p$$

Proof. By definition a constraint store that is input to constraint propagation has a finite set of values and variables [1]. This implies that the set of variables $var(p)$ is a finite set where $|var(p)| \in \mathbb{N}$ and that the set of values $s(x)$ for all variables $x \in var(p)$ is a finite set where $|s(x)| \in \mathbb{N}$.

A correct propagator is contracting, this implies that if $p^n(s)$ is not idempotent then $p^{n+1}(s) < p^n(s)$. I.e $p^{n+1}(s)$ is strictly stronger than $p^n(s)$, which means that $\exists x \in var(p)$ where $p^{n+1}(s)(x) \subset p^n(s)(x)$.

A propagator p must by definition be idempotent on a store s if $s(x) = \emptyset \quad \forall x \in var(p)$, i.e $p(s) = s$.

Now we can see that $p^n(s)$ for arbitrary natural number n and constraint store s computes a sequence of stores $s > s_1 > s_2 > s_3 > \cdots > s_{n-1}$ that is a well founded order (this is a property of the $>$ relation on stores[1]).

$$\implies \exists n \in \mathbb{N} \text{ such that } p^n \text{ is idempotent for any store } s \text{ and correct propagator } p$$

□

The idempotent property is not true for arbitrary functions on arbitrary sets. A counterexample is a function f on sets that is not contracting but ever expanding by adding a natural number to the set.

Theorem 0.4. *Arbitrary functions on arbitrary sets are not always idempotent*

Proof. By construction of a counterexample.

$$f(s) = s \cup \{x\} \land x \in \mathbb{N} \setminus s$$

$$\lim_{x \to \infty} |f(x)| = \infty$$

□

References

[1] Christian Schulte. Course notes constraint programming (id2204) vt 2012, 2012. [Online; accessed 19-April-2017].