A parametric propagator for discretely convex pairs of sum constraints[1]

Jean-Noël Monette*, Nicolas Beldiceanu**, Pierre Flener*, and Justin Pearson*

*ASTRA Research Group on Constraint Programming
Uppsala University

**TASC Team (CNRS/INRIA)
Mines de Nantes

CP 2013

[1] supported by grants 2011-6133 and 2012-4908 of VR, the Swedish Research Council

Motivation

- SUM constraints are widely used.
- Alone: not much propagation.
- With other constraints: better propagation is possible, if considered together.

Example: DEVIATION(x, Δ, μ)

$$\sum_{i \in [1,n]} |x_i - \mu| \leq \Delta$$

$$\sum_{i \in [1,n]} x_i / n = \mu$$

[Schaus et al., CP 2007] propose a bounds-consistency propagator that runs in $\mathcal{O}(n)$ time.

UPPSALA UNIVERSITET

Example: DEVIATION(x, Δ, μ)

$$\sum_{i \in [1,n]} |n \cdot x_i - n \cdot \mu| \leq \Delta$$

$$\sum_{i \in [1,n]} x_i = n \cdot \mu$$

[Schaus et al., CP 2007] propose a bounds-consistency propagator that runs in $\mathcal{O}(n)$ time.

Example: SPREAD(x, Δ, μ)

$$\sum_{i \in [1,n]} (n \cdot x_i - n \cdot \mu)^2 \leq \Delta$$

$$\sum_{i \in [1,n]} x_i = n \cdot \mu$$

[Schaus and Regin, to appear] propose a bounds-consistency propagator that runs in $\mathcal{O}(n \log n)$ time.

Example: Linear Inequality and Counting

$$\sum_{i \in [1,n]} a_i \cdot x_i \leq b$$

$$\sum_{i \in [1,n]} [x_i = v] = c$$

[Razakarison et al., SoCS 2013] propose a domain-consistency propagator that runs in $\mathcal{O}(n \cdot (\log n + p))$ time (where p is the complexity of filtering one variable).

Motivation
Pair of Sums
Solving
Results
Conclusion

The general case

$$\sum_{i \in [1,n]} f_i(x_i) \leq \overline{f}$$

$$\underline{g} \leq \sum_{i \in [1,n]} g_i(x_i) \leq \overline{g}$$

UPPSALA UNIVERSITET

The general case: Reformulation

$$\sum_{i \in [1,n]} f_i(x_i) \leq \overline{f}$$

$$\underline{g} \leq \sum_{i \in [1,n]} g_i(x_i) \leq \overline{g}$$

Reformulate the problem to simplify its presentation and bring out some special structure.

Motivation
Pair of Sums
Solving
Results
Conclusion

The general case: Reformulation

$$\sum_{i \in [1,n]} f_i(x_i) \leq \overline{f}$$

$$\underline{g} \leq \sum_{i \in [1,n]} g_i(x_i) \leq \overline{g}$$

Motivation

Pair of Sums

Solving

Results

Conclusion

The general case: Reformulation

$$\sum_{i \in [1,n]} f_i(x_i) \leq \overline{f}$$

$$\underline{g} \leq \sum_{i \in [1,n]} g_i(x_i) \leq \overline{g}$$

1. Introduce $y_i = g_i(x_i)$

UPPSALA UNIVERSITET

The general case: Reformulation

$$\sum_{i \in [1,n]} \min f_i(g_i^{-1}(y_i)) \leq \overline{f}$$

$$\underline{g} \leq \sum_{i \in [1,n]} y_i \leq \overline{g}$$

1. Introduce $y_i = g_i(x_i)$

UPPSALA UNIVERSITET

The general case: Reformulation

$$\sum_{i \in [1,n]} \min f_i(g_i^{-1}(y_i)) \leq \overline{f}$$

$$\underline{g} \leq \sum_{i \in [1,n]} y_i \leq \overline{g}$$

1. Introduce $y_i = g_i(x_i)$
2. Define $h_i(v) = \min f_i(g_i^{-1}(v))$

The general case: Reformulation

$$\sum_{i \in [1,n]} h_i(y_i) \leq \overline{f}$$

$$\underline{g} \leq \sum_{i \in [1,n]} y_i \leq \overline{g}$$

1. Introduce $y_i = g_i(x_i)$
2. Define $h_i(v) = \min f_i(g_i^{-1}(v))$

The general case: Reformulation

$$\sum_{i \in [1,n]} h_i(y_i) \leq \overline{f}$$

$$\underline{g} \leq \sum_{i \in [1,n]} y_i \leq \overline{g}$$

1. Introduce $y_i = g_i(x_i)$
2. Define $h_i(v) = \min f_i(g_i^{-1}(v))$
3. Define $H(b) =$

$$\min \left\{ \sum_{i \in [1,n]} h_i(y_i) \;\middle|\; \sum_{i \in [1,n]} y_i = b \wedge \forall i \in [1,n] : y_i \in g_i(\mathrm{D}_{x_i}) \right\}$$

UPPSALA UNIVERSITET

The general case: Reformulation

$$H(z) \leq \overline{f}$$
$$\underline{g} \leq z \leq \overline{g}$$

1. Introduce $y_i = g_i(x_i)$
2. Define $h_i(v) = \min f_i(g_i^{-1}(v))$
3. Define $H(b) =$
$$\min \left\{ \sum_{i \in [1,n]} h_i(y_i) \,\middle|\, \sum_{i \in [1,n]} y_i = b \wedge \forall i \in [1,n] : y_i \in g_i(\mathrm{D}_{x_i}) \right\}$$
4. Introduce $z = \sum_{i \in [1,n]} y_i$.

The general case: Reformulation

$$H(z) \leq \overline{f}$$
$$\underline{g} \leq z \leq \overline{g}$$

1. Introduce $y_i = g_i(x_i)$
2. Define $h_i(v) = \min f_i(g_i^{-1}(v))$
3. Define $H(b) =$
$$\min \left\{ \sum_{i \in [1,n]} h_i(y_i) \;\middle|\; \sum_{i \in [1,n]} y_i = b \wedge \forall i \in [1,n] : y_i \in g_i(\mathrm{D}_{x_i}) \right\}$$
4. Introduce $z = \sum_{i \in [1,n]} y_i$.

The reformulated problem is equivalent to the original one.

UPPSALA UNIVERSITET

Motivation

Pair of Sums

Solving

Results

Conclusion

Examples of functions

Common Name	$f_i(u)$	$g_i(u)$	$h_i(v)$
LINEAR	$a_i \cdot u$	0	$\begin{cases} a_i \cdot \min \mathrm{D}_{x_i} & \text{if } a_i > 0 \\ a_i \cdot \max \mathrm{D}_{x_i} & \text{if } a_i \leq 0 \end{cases}$
WEIGHTEDAVERAGE	$a_i \cdot u$	u	$a_i \cdot v$
DEVIATION	$\lvert n \cdot u - n \cdot \mu \rvert$	u	$\lvert n \cdot v - n \cdot \mu \rvert$
SPREAD	$(n \cdot u - n \cdot \mu)^2$	u	$(n \cdot v - n \cdot \mu)^2$
L_p-NORM	$\lvert n \cdot u - n \cdot \mu \rvert^p$	u	$\lvert n \cdot v - n \cdot \mu \rvert^p$
LINEAR and COUNT	$a_i \cdot u$	$[u \in \mathcal{V}]$	$\begin{cases} a_i \cdot \min(\mathrm{D}_{x_i} \setminus \mathcal{V}) & \text{if } v = 0 \wedge a_i > 0 \\ a_i \cdot \max(\mathrm{D}_{x_i} \setminus \mathcal{V}) & \text{if } v = 0 \wedge a_i \leq 0 \\ a_i \cdot \min(\mathrm{D}_{x_i} \cap \mathcal{V}) & \text{if } v = 1 \wedge a_i > 0 \\ a_i \cdot \max(\mathrm{D}_{x_i} \cap \mathcal{V}) & \text{if } v = 1 \wedge a_i \leq 0 \end{cases}$
MODANDDIV ($a_i > 0$)	$u - a_i \cdot \lfloor u/a_i \rfloor$	$\lfloor u/a_i \rfloor$	$\max(0, \min \mathrm{D}_{x_i} - a_i \cdot v)$

Checking for Satisfiability

$$\begin{aligned} \text{minimise} \quad & H(z) \\ \text{such that} \quad & \underline{g} \leq z \leq \overline{g} \end{aligned} \tag{1}$$

where

$$H(b) = \min \left\{ \sum_{i \in [1,n]} h_i(y_i) \;\middle|\; \sum_{i \in [1,n]} y_i = b \wedge \forall i \in [1,n] : y_i \in g_i(\mathrm{D}_{x_i}) \right\}$$

Theorem

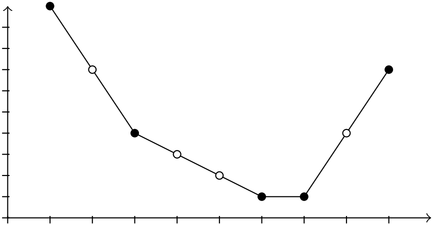
Problem (1) can be solved greedily if each h_i is discretely convex.

Discrete Convexity

Definition (Discrete Convexity)

A function $f\colon A \to B$, where $A, B \subseteq \mathbb{Z}$, is *discretely convex* if

- A is an interval, and
- $\forall v \in A\colon (v-1) \in A \wedge (v+1) \in A \Rightarrow$
  $2 \cdot f(v) \leq f(v-1) + f(v+1)$.



Properties

Theorem

Problem (1) can be solved greedily if each h_i is discretely convex.

Properties

- $H(.)$ is also discretely convex.
- $H(.)$ can be constructed in polynomial time from the h_i.
- $\min H(z)$ s.t. $y_j = v$ for some j can be computed incrementally from v to $v + 1$ (used for filtering).

Propagator

1. Construct $H(.)$ and find its minimum (two ways)
2. Filter each variable (two ways)

The algorithms are in the paper.

UPPSALA UNIVERSITET

Motivation
Pair of Sums
Solving
Results
Conclusion

Propagator

1. Construct $H(.)$ and find its minimum (two ways)
2. Filter each variable (two ways)

The algorithms are in the paper.

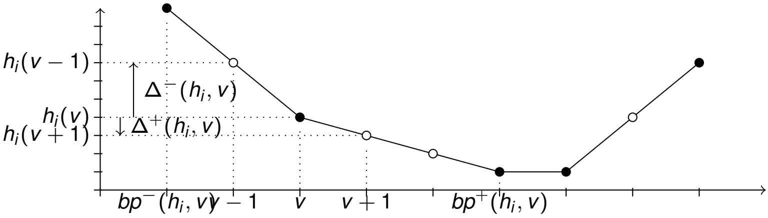
Time complexity of the different versions

Version	Time complexity
Traversing, general case	$\mathcal{O}(n \cdot (s(h) \cdot p + s(H) + r(h) \cdot c)$
Sorting, general case	$\mathcal{O}(n \cdot (s(h) \cdot p + s(h) \cdot \log(n \cdot s(h)) + r(h) \cdot c))$
Traversing, special case	$\mathcal{O}(n \cdot (s(h) \cdot p + s(H) + c))$
Sorting, special case	$\mathcal{O}(n \cdot (s(h) \cdot p + s(h) \cdot \log(n \cdot s(h)) + s(H) + c))$

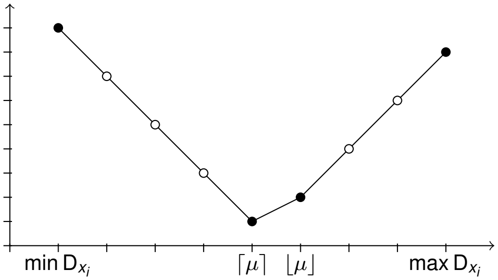
Parametric algorithms

Parameters to instantiate

Functions	Procedures
$\text{argmin}_{v \in g_i(D_{x_i})} h_i(v)$	$\text{FILTER}(g_i(x_i) \leq v)$
$\Delta^+(h_i, v)$	$\text{FILTER}(g_i(x_i) \geq v)$
$\Delta^-(h_i, v)$	$\text{FILTER}(g_i(x_i) = v \Rightarrow f_i(x_i) \leq u)$
$\text{bp}^+(h_i, v)$	
$\text{bp}^-(h_i, v)$	



Example: Parameters for DEVIATION



Parameters to instantiate

Functions	Procedures
$\text{argmin}_{v \in g_i(D_{x_i})} h_i(v)$	$\text{FILTER}(g_i(x_i) \leq v)$
$\Delta^+(h_i, v)$	$\text{FILTER}(g_i(x_i) \geq v)$
$\Delta^-(h_i, v)$	$\text{FILTER}(g_i(x_i) = v \Rightarrow f_i(x_i) \leq u)$
$\text{bp}^+(h_i, v)$	
$\text{bp}^-(h_i, v)$	

Example: Parameters for DEVIATION

Parameter	Instantiation
$\mathrm{argmin}_{v \in g_i(\mathrm{D}_{x_i})} h_i(v)$	$\begin{cases} \lceil\mu\rceil & \text{if } \min \mathrm{D}_{x_i} \leq \mu \leq \max \mathrm{D}_{x_i} \wedge \lceil\mu\rceil - \mu < \mu - \lfloor\mu\rfloor \\ \lfloor\mu\rfloor & \text{if } \min \mathrm{D}_{x_i} \leq \mu \leq \max \mathrm{D}_{x_i} \wedge \lceil\mu\rceil - \mu \geq \mu - \lfloor\mu\rfloor \\ \min \mathrm{D}_{x_i} & \text{if } \mu < \min \mathrm{D}_{x_i} \\ \max \mathrm{D}_{x_i} & \text{if } \mu > \max \mathrm{D}_{x_i} \end{cases}$
$\Delta^+(h_i, v)$	$\begin{cases} +\infty & \text{if } v = \max \mathrm{D}_{x_i} \\ -n & \text{if } v < \lfloor\mu\rfloor \\ n \cdot (\lceil\mu\rceil + \lfloor\mu\rfloor) - 2 \cdot n \cdot \mu & \text{if } v = \lfloor\mu\rfloor \wedge \lfloor\mu\rfloor \neq \lceil\mu\rceil \\ n & \text{if } v \geq \lceil\mu\rceil \end{cases}$
$\mathrm{bp}^+(h_i, v)$	$\begin{cases} +\infty & \text{if } v = \max \mathrm{D}_{x_i} \\ \min(\max \mathrm{D}_{x_i}, \lfloor\mu\rfloor) & \text{if } v < \lfloor\mu\rfloor \\ \lceil\mu\rceil & \text{if } v = \lfloor\mu\rfloor \wedge \lfloor\mu\rfloor \neq \lceil\mu\rceil \\ \max \mathrm{D}_{x_i} & \text{if } v \geq \lceil\mu\rceil \end{cases}$
FILTER$(g_i(x_i) \leq v)$	FILTER$(x_i \leq v)$
FILTER$(g_i(x_i) = v \Rightarrow f_i(x_i) \leq u)$	FILTER$(\lvert n \cdot v - n \cdot \mu \rvert > u \Rightarrow x_i \neq v)$

A Few Other Instantiations

Constraint	Consistency	Complexity	Literature
LINEAR	bounds($\mathbb{Z}$)	$\mathcal{O}(n \cdot \log n)$	$\mathcal{O}(n)$
WEIGHTEDAVERAGE	bounds($\mathbb{Z}$)	$\mathcal{O}(n \cdot \log n)$	$\mathcal{O}(n)$
DEVIATION	bounds($\mathbb{Z}$)	$\mathcal{O}(n)$	$\mathcal{O}(n)$
SPREAD	bounds($\mathbb{Z}$)	$\mathcal{O}(n \cdot d)$	$\mathcal{O}(n \cdot \log n)$
L_p-NORM	bounds($\mathbb{Z}$)	$\mathcal{O}(n \cdot d)$	-
LINEAR and COUNT	domain	$\mathcal{O}(n \cdot (\log n + p + c))$	$\mathcal{O}(n \cdot (\log n + p + c))$

n: number of variables
d: size of the union of domains
p, c: complexity of the parameters

Experiments

Compare to existing implementations:

Solving the Balanced Academic Curriculum Problem in OscaR.

DEVIATION as an objective: 7% slower with our propagator

SPREAD as an objective: 28% faster with our propagator

Code available at

http://www.it.uu.se/research/group/astra/software/convexpairs

Conclusion

- Discrete convexity
- Parametric propagator
- Efficient
- Extensions:
  - Generating the parameters
  - more than 2 sums
  - incrementality
  - ...