A Box-Consistency Contraction Operator Based on Extremal Functions

Gilles Trombettoni, Yves Papegay, Gilles Chabert, Odile Pourtallier
COPRIN project and ENSIETA
INRIA, 2004 route des Lucioles
06902 Sophia-Antipolis Cedex, France
trombe@sophia.inria.fr, papegay@sophia.inria.fr, gilchab@gmail.com,
Odile.Pourtallier@sophia.inria.fr

Abstract

Interval-based solving techniques use different operators to compute, in a reliable way, the real solutions of a system of equations. Some *contraction* operators come from the constraint programming community. Starting from an initial search interval for every unknown, contraction operators reduce these intervals and finally converge onto a *box* that contains all the solutions of the system. The state-of-the-art operators `HC4` [1] and `Box-consistency` [1] consider a single equation of the system at each step, and propagate the obtained reductions in the rest of the system until no interval can be reduced.

We consider a particular equation $f(a, x) = 0$ of a given system. The $n-1$ first variables of $f : \mathbb{R}^{n-1} \times \mathbb{R} \rightarrow \mathbb{R}$ have been agregated into a vectorial variable $a \in \mathbb{R}^{n-1}$. `BoxRevise` (also known as BoxNarrow) is the atomic procedure used by `Box-consistency`. Starting from an initial interval $[x]$ for x, `BoxRevise` computes a reduced interval $[l, r]$ for x with no loss of solution (i.e., $\forall x_s \in [x]\ s.t.\ x_s \notin [l, r], \forall a_s \in [a] : f(a_s, x_s) \neq 0$, $[a]$ being the initial box of a). In practice, `BoxRevise` works with $f([a], x)$, the multivalued function defined on a single variable x obtained by replacing a by the box/interval $[a]$. It computes l (respectively r) as the leftmost (respectively the rightmost) root of $f([a], x) = 0$.

Following the latest version `BC4` [1] of `Box-consistency`, if the analytic expression $f(a, x)$ contains only one occurrence of x, the well-known `HC4Revise` narrowing operator [1] computes $[l, r]$ very quickly. Otherwise, `BoxRevise` calls iteratively bisection steps on $[x]$ and univariate interval Newton to compute l (respectively r). This process may converge slowly because $f([a], x)$ is a *multivalued* ("thick") function.

The `PolyBox` (*polynomial Box-consistency*) operator proposed in this paper implements a more efficient `BoxRevise` procedure when $f([a], x)$ satisfies some conditions. These conditions apply for example when $f([a], x)$

is a polynomial. We focus on the polynomial case in this abstract. `PolyBox` first performs symbolic manipulations to rewrite $f([a], x)$ as $g_{[a]}(x) = \sum_{i=0}^{i=d} f_i([a]).x^i$. This step is crucial since the "thickness" of $f([a], x)$ depends on its symbolic form.

We define the *extremal functions* as $\underline{g_{[a]}}(x) = \min_{a_s \in [a]} f(a_s, x)$ and $\overline{g_{[a]}}(x) = \max_{a_s \in [a]} f(a_s, x)$, and we show that these two functions are piecewise polynomial functions, with coefficients in $\{\overline{f_i([a])}, \underline{f_i([a])}\}$. For instance, if $g_{[a]}(x) = [-2, 3]x^2 + [-4, -2]x + [4, 5]$, then $\overline{g_{[a]}}(x) = g^+(x)$ for $x \geq 0$, and $\overline{g_{[a]}}(x) = g^-(x)$ for $x \leq 0$, with $g^+(x) = 3x^2 - 2x + 5$ and $g^-(x) = 3x^2 - 4x + 5$. The key point is to work with these two (univalued) functions $\underline{g_{[a]}}$ and $\overline{g_{[a]}}$ instead of $g_{[a]}$ (multivalued).

Let us describe the determination of l. The evaluation of $\overline{g_{[a]}}$ and $\underline{g_{[a]}}$ on $\underline{[x]}$ determines with which extremal function to work, e.g., with $\overline{g_{[a]}}$. According to the degree d of the polynomial, the roots of $\overline{g_{[a]}}(x) = 0$ are then determined either analytically ($d \leq 4$), or numerically ($d \geq 5$). For $d \geq 5$, we use `BoxRevise` which converges quickly since it is applied to a univalued function.

This idea has been described in two pages by Van hentenryck et al. in [2] and implemented in the solver Numerica. We go beyond this raw idea, present in detail a new implementation of it and provide a full experimental evaluation of the approach that shows the efficiency of an algorithm based on extremal functions. `PolyBox` has been implemented in `Mathematica` and in the interval-based solver `Ibex` in `C++`.

References:

[1] F. Benhamou, F. Goualard, L. Granvilliers, J.-F. Puget; *Revising hull and box consistency* Proc. of the International Conference on Logic Programming, ICLP 1999, pp. 230-244

[2] P. Van Hentenryck, D. Mc Allester, and D. Kapur; *Solving Polynomial Systems Using a Branch and Prune Approach.* SIAM Journal on Numerical Analysis , 34(2), 1997.

Keywords: Interval analysis, numerical CSPs, contraction, Box consistency, extremal functions