# On the convergence of feasibility based bounds tightening

Pietro Belotti

*Dept. of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA, USA*

Sonia Cafieri

*Dept. Mathématiques et Informatique, ENAC, 7 av. E. Belin, 31055 Toulouse, France*

Jon Lee

*Dept. of Mathematical Sciences, IBM T.J. Watson Research Center, PO Box 218, Yorktown Heights, NY 10598, USA*

Leo Liberti *

*LIX, École Polytechnique, 91128 Palaiseau, France*

*Key words:* spatial Branch-and-Bound, range reduction, lattice, fixed point, MINLP, global optimization, constraint programming.

## 1 Introduction

Global Optimization and Mixed-Integer Nonlinear Programming problems such as $\min\{f(x) \mid g^L \le g(x) \le g^U \wedge x^L \le x \le x^U \wedge \forall j \in Z\ (x_j \in \mathbb{Z})\}$, where $f : \mathbb{R}^n \to \mathbb{R}$, $g : \mathbb{R}^n \to \mathbb{R}^m$, $g^L, g^U \in \mathbb{R}^m$, $x^L, x, x^U \in \mathbb{R}^n$ and $Z \subseteq \{1, \ldots, n\}$,

⋆ Financial support by grants System@tic "EDONA", ANR 07-JCJC-0151 "Ars", ANR 08-SEGI-023 "Asopt", Digiteo Emergence "Paso" is gratefully acknowledged.

* Corresponding author.

*Email addresses:* `belotti@lehigh.edu` (Pietro Belotti), `sonia.cafieri@enac.fr` (Sonia Cafieri), `jonlee@us.ibm.com` (Jon Lee), `liberti@lix.polytechnique.fr` (Leo Liberti).

are usually solved to $\varepsilon$-guaranteed approximation by the spatial Branch-and-Bound (sBB) algorithm [2], a variant of the usual Branch-and-Bound for dealing with nonlinear, possibly nonconvex $f, g$. Since the gap between the original problem $P$ and its convex relaxation $\bar{P}$ is due both to integral variable restrictions being lifted as well as nonconvex functions being replaced by a convex relaxation, sBB is able to branch at continuous variables as well as integer ones. If $\bar{x}$ solves $\bar{P}$, the standard disjunction used at a node in the sBB search tree is $x_j \leq \bar{x}_j \vee x_j \geq \bar{x}_j$, the more usual one $x_j \leq \lfloor \bar{x}_j \rfloor \vee x_j \geq \lceil \bar{x}_j \rceil$ being used only if $j \in Z$.

At any sBB node, it is important to make sure that the variable ranges $x^L \leq x \leq x^U$ for that node are as tight as the constraint restrictions $g^L \leq g(x) \leq g^U$ allow. Letting $\mathcal{F}(P)$ be the feasible region of $P$, we would wish to replace $X^0 = [x^L, x^U]$ with $\tilde{X} = [\tilde{x}^L, \tilde{x}^U]$ such that $\tilde{x}^L_i = \min_{x \in \mathcal{F}(P)} x_i$ and $\tilde{x}^U_i = \max_{x \in \mathcal{F}(P)} x_i$ for all $i \leq n$. Since these $2n$ problems are as hard as $P$, we relax these requirements. There are two standards relaxations: Optimization Based Bounds Tightening (OBBT) [3,2] and Feasibility Based Bounds Tightening (FBBT) [1,2]. The former consists in replacing $\mathcal{F}(P)$ with $\mathcal{F}(\bar{P})$. The latter, whose convergence properties are the object of this paper, is also known in Constraints Programming as a range reduction device. FBBT relies on interval arithmetic to derive the constraint ranges $\bar{G} = [\bar{g}^L, \bar{g}^U]$ implied by the variable ranges $X^0$ at a given sBB node; if $\bar{G} \supsetneq G^0 = [g^L, g^U]$, FBBT uses inverse interval arithmetic to propagate $G^0$ back to tightened variable ranges $X'$. This basic FBBT step is iterated until convergence, generating an interval sequence $X^0, X^1, \ldots$. Since OBBT is usually slower than FBBT, OBBT is only applied at the root sBB node and FBBT is applied at each node. Furthermore, to simplify inverse interval arithmetic, FBBT often only considers a subset of linear constraints in $g$. We shall therefore make the assumption — without excessive loss of generality — that the symbol $g$ denotes the linear constraints of $P$, which we denote as $g^L \leq Ax \leq g^U$ for some $m \times n$ matrix $A$.

The main trouble with FBBT is that its worst-case running time is infinite in the size of its input $(m, n, X^0, A, G^0)$. For example, on the instance $(2, 2, ([0,1],[0,1]), \left(\begin{smallmatrix} a & -1 \\ 1 & -a \end{smallmatrix}\right), ([0,0],[0,0]))$, FBBT yields the infinite interval sequence $([0, 1/a^{2k-1}], [0, 1/a^{2k}])$ whenever $a > 1$. Enforcing finite convergence by terminating at the first iteration $k$ such that $\mathscr{L}(X^{k-1} \triangle X^k) \leq \varepsilon$, where $\varepsilon > 0$ is given and $\mathscr{L}$ is the Lebesgue measure in $\mathbb{R}$, yields a finite but unbounded worst-case time complexity: given a fixed iteration bound $K$ there are always instances where the FBBT takes longer than $K$ iterations to reach the $\varepsilon$ termination condition (it suffices to decrease the value $a$ appropriately). In practice, such occurrences are far from rare, specially when the coefficients of $Ax$ are obtained by previous floating point operations, which might cause a small but positive $|a_i - a_j|$ even if $a_i, a_j$ are supposed to be equal.

In this paper we propose a new method for finding the limit point of the FBBT sequence in polynomial time, based solving a Linear Program (LP) modelling the greatest fixed point of the FBBT in the interval lattice.

## 2 Fixed points in the interval lattice

A *lattice* is a set $\Lambda$ partially ordered by the relation $\sqsubseteq$ endowed with two operations $\sqcup$ (*join*), $\sqcap$ (*meet*) such that $x \sqsubseteq x \sqcup y$, $y \sqsubseteq x \sqcup y$ and $x \sqcap y \sqsubseteq x$, $x \sqcap y \sqsubseteq y$. A lattice is *complete* if there exist elements $\bot, \top$ such that $\bot \sqsubseteq x \sqsubseteq \top$ for all $x \in \Lambda$. An operator $F : \Lambda \to \Lambda$ is *monotone* if $x \sqsubseteq y$ implies $F(x) \sqsubseteq f(y)$ and *deflationary* if $F(x) \sqsubseteq x$ for all $x \in \Lambda$. The set of all real intervals forms a lattice $\mathscr{I}$ under set inclusion $\subseteq$, with set intersection $\cap$ as meet and interval union (smallest interval including two intervals) $\cup$ as join. The lattice structure is extended to arrays of intervals in the standard way. In the rest of the paper, we let $X$ be the interval vector $(X_1, \ldots, X_n) \in \mathscr{I}^n$ and $G = (G_1, \ldots, G_m) \in \mathscr{I}^m$.

FBBT consists of two phases: upwards and downwards propagation. We define $\mathsf{up} : \mathscr{I}^n \to \mathscr{I}^m$ and $\mathsf{down} : \mathscr{I}^m \to \mathscr{I}^n$ as:

$$\mathsf{up}(X) = (G_i^0 \cap \sum_{j \le n} a_{ij} X_j \mid i \le m) \tag{1}$$

$$\mathsf{down}(G) = \bigcap_{i \le m} (X_j \cap \frac{1}{a_{ij}} (G_i - \sum_{\ell \ne j} a_{i\ell} X_\ell) \mid j \le n), \tag{2}$$

where all arithmetic operators have interval semantics [5]. We now define the FBBT iteration as an operator $\mathsf{fbbt} : \mathscr{I}^n \to \mathscr{I}^n$ by $\mathsf{fbbt}(X) = \mathsf{down}(\mathsf{up}(X \cap X^0))$ (we remark that our definition of $\mathsf{fbbt}$ depends on the initial interval vector $X^0$). Because all linear interval arithmetic and lattice operators are monotone [5] and the composition of monotone operators is monotone [6], $\mathsf{fbbt}$ is a monotone operator. Furthermore, because of the intersections, intervals are changed only if the $\mathsf{up}$ and $\mathsf{down}$ actions make them smaller. Again, the composition of deflationary operators is deflationary [6]: hence $\mathsf{fbbt}$ is deflationary. By applying Thm. 12.9 in [6] to the dual lattice obtained by inverting $\top$ and $\bot$, $\sqsubseteq$ and $\sqsupseteq$, meet and join, we have that the sequence $(\mathsf{fbbt}^k(X) \mid k \ge 0)$ converges to the greatest fixed point (gfp) of $\mathsf{fbbt}$, i.e. the largest (in the lattice order) interval vector $X$ such that $\mathsf{fbbt}(X) = X$. In other words $\mathrm{gfp}(\mathsf{fbbt}) = \sup\{X \mid X = \mathsf{fbbt}(X)\}$. By Tarski's Fixed Point Theorem [7], equality can be replaced with $\subseteq$. Furthermore, the operator $|\cdot| : \mathscr{I}^n \to \mathbb{R}$ given by $|X| = \sum_{j \le n} (x_j^U - x_j^L)$ is monotone with the lattice order; since the lattice is complete, we obtain:

$$\mathrm{gfp}(\mathsf{fbbt}) = \mathrm{argmax}\{|X| \mid X \subseteq \mathsf{fbbt}(X)\}, \tag{3}$$

which we state as the following "interval linear problem" with parameters

$(m, n, X^0, A, G^0)$ and interval decision variable arrays $X, G$:

$$\max\{|X| \mid X \subseteq X^0 \wedge G \subseteq \mathsf{up}(X) \wedge X \subseteq \mathsf{down}(G)\}, \tag{4}$$

It is possible to write an ordinary LP whose optimal solution is the same as (4).

## 3 Computational validation

By way of preliminary computational validation of our approach, we solved four significant instances using CPLEX 11.0 [4] on an Intel Core 2 Duo 1.4GHz with 3GB RAM running Linux. For each instance we record the gfp, the seconds of user CPU time to solution with either method, the absolute error $E = \sum_{j \le n} \mathscr{L}(X^* \triangle \mathrm{gfp}(\mathsf{fbbt}))$ where $X^*$ is the output of either method ($\varepsilon_{\mathrm{FBBT}} = 10^{-6}$), and the relative error $R$ of the FBBT obtained by letting it run only for as long as the LP method takes to converge to the (precise) gfp's.

| Instance | $x_1 - 1.01x_2 = 0$<br>$-1.01x_1 + x_2 = 0$<br>$x_1, x_2 \in [-1, 1]$ | $x_1 - 1.01x_2 = 1$<br>$-1.01x_1 + x_2 = -1.01$<br>$x_1, x_2 \in [-1, 1]$ | $x_1 - 1.01x_2 = 0$<br>$-1.01x_1 + x_2 = 0$<br>$x_3 + 100x_1 \geq -100$<br>$x_1, x_2 \in [-1, 1]$, $x_3 \in [-10, 1]$ | $x_1 - 1.01x_2 = -1.01$<br>$-1.01x_1 + x_2 = 1$<br>$x_3 + 100x_1 \geq -100$<br>$x_1, x_2 \in [-1, 1]$, $x_3 \in [-10, 1]$ |
|---|---|---|---|---|
| gfp | ([0,0],[0,0]) | ([1,1],[0,0]) | ([0,0],[0,0],[-10,0]) | ([0,0],[1,1],[-10,0]) |
| $\mathrm{CPU}_{\mathrm{FBBT}}$ | 0.04 | 0.04 | 0.06 | 0.06 |
| $\mathbf{CPU_{LP}}$ | **0** | **0** | **0.004** | **0.004** |
| $E_{\mathrm{FBBT}}$ | 1e-4 | 5e-5 | 1.5e-6 | 5e-5 |
| $\boldsymbol{E_{\mathrm{LP}}}$ | **0** | **0** | **0** | **0** |
| $R_{\mathrm{FBBT}}$ | 2.29 | 1.47 | 2.33 | 1.21 |

## References


[1] D.E. Andersen and K.D. Andersen. Presolving in linear programming. *Mathematical Programming*, 71:221–245, 1995.

[2] P. Belotti, J. Lee, L. Liberti, F. Margot, and A. Wächter. Branching and bounds tightening techniques for non-convex MINLP. *Optimization Methods and Software*, 24(4):597–634, 2009.

[3] A. Caprara and M. Locatelli. Global optimization problems and domain reduction strategies. *Mathematical Programming*, to appear.

[4] ILOG. *ILOG CPLEX 11.0 User's Manual*. ILOG S.A., Gentilly, France, 2008.

[5] R.E. Moore, R.B. Kearfott, and M.J. Cloud. *Introduction to Interval Analysis*. SIAM, Philadelphia, 2009.

[6] S. Roman. *Lattices and Ordered Sets*. Springer, New York, 2008.

[7] A. Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5(2):285–309, 1955.