1D4-OS-11a-3

Size complexity of BDD construction of Pseudo-Boolean constraints in binary/mixed-radix base form

Naoki Nagatsuka[*1] Masahiko Sakai[*1] Harald Zankl[*2] Keiichirou Kusakari[*3]

[*1] Graduate School of Information Science, Nagoya University
[*2] Institute of Compute Science, University of Innsbruck [*3] Faculty of Engineering, Gifu University

An ROBDD with ascending variable order representing a Pseudo-Boolean constraint has polynomial size if all coefficients in the constraint are powers of two (Abío et al. 2012). This paper extends the result to descending variable-orders and generalizes it to Pseudo-Boolean constraints having mixed-radix base coefficients (for ascending and descending variable-orders). We implemented the proposed constructions and report on experimental results.

1. Introduction

Pseudo-Boolean (PB) constraints are conjunctions of linear inequalities over Boolean variables. Several kinds of solvers have been developed, see e.g. `http://www.cril.univ-artois.fr/PB12/` for a comparison. Typical approaches to solve PB constraints employ *Integer Linear Programming* (restricted to 0-1 variables), DPLL procedures (regarding PB constraints as generalized clauses [6]), as well as transformations of PB constraints to CNF (via adders, sorting networks, and BDDs [2, 5]).

In [1], Abío et al. have shown that a PB constraint where all coefficients are powers of two admits a polynomial sized ROBDD with ascending variable-order, i.e., variables having smaller coefficients are placed closer to the root. Hence, performing a binary expansion of the coefficients in a PB constraint as a pre-processing step yields a polynomial sized ROBDD. For example, a PB constraint $2x+3y \leq 3$ is transformed to $2x+2y+y' \leq 3$ and $y = y'$ by binary expansion. In this way, PB constraints can be converted into an equi-satisfiable and polynomial sized CNF via ROBDDs.

Codish et al. proposed the notion of *optimal-base decomposition* of a PB constraint, which is a minimal length representation with a mixed-radix base expansion of coefficients [4].

This paper extends the result of [1] to ROBDDs with descending variable-order and shows that the ROBDD from a mixed-radix base expanded PB constraint is also of polynomial size (for ascending and descending variable-orders).

We show experimental results of a MiniSat+ based solver, in which we incorporated the proposed BDD construction.

2. PB constraints and ROBDDs

A PB constraint is of the form $a_1x_1 + \cdots + a_nx_n \leq K$, where the a_i's and K are integers such that $a_i > 0$ and the x_i's are Boolean variables. Since PB constraints resemble Boolean functions, *Binary Decision Diagrams* (BDDs) may represent PB constraints. Let C be the PB constraint

Contact: Naoki Nagatsuka, Nagoya University, C3-1(631) Furo-cho, Chikusa-ku Nagoya, Japan, and nagatsuka@trs.cm.is.nagoya-u.ac.jp

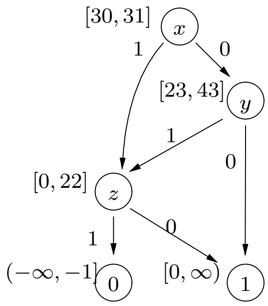

Figure 1: ROBDD of $9x + 21y + 23z \leq 30$

$a_1x_1 + \cdots + a_nx_n \leq K$. We say $[\beta, \gamma]$ is the *interval of* C if for $M \in [\beta, \gamma]$, i.e., $\beta \leq M \leq \gamma$, $a_1x_1 + \cdots + a_nx_n \leq M$ and C are equivalent (seen as Boolean functions) [1]. For a PB constraint $a_1x_1 + \cdots + a_nx_n \leq K$, a variable-order is called *ascending* if $x_i < x_j$ implies $a_i \leq a_j$ for all i, j. Similarly, it is called *descending* if $x_i < x_j$ implies $a_i \geq a_j$.

Example 1 *A BDD for* $9x + 21y + 23z \leq 30$ *with the ascending order* $x < y < z$ *is shown in Figure 1. This is also an ROBDD.*

Here *ROBDDs* are a canonical representation for Boolean functions under a given variable order [3]. For an ROBDD, every pair of nodes represents different Boolean functions.

Note that a sub-graph of an ROBDD also is an ROBDD. For example, the node in Figure 1 with *selector variable* y represents $21y + 23z \leq M$ for any $M \in [23, 43]$. The following propositions state properties used later on where we assume that the ROBDD represents a PB constraint $a_1x_1 + \cdots + a_nx_n \leq K$.

Proposition 2 ([1]) *If* $[\beta, \gamma]$ *is the interval of a node* ν *in an ROBDD with selector variable* x_i *then:*

(i) For each $i \in \{1, \ldots, n\}$ *an assignment* $\{x_j = v_j\}_{j=i}^{n}$ *exists with* $a_iv_i + \cdots + a_nv_n = \beta$.

(ii) For each $i \in \{1, \ldots, n\}$ *an assignment* $\{x_j = v_j\}_{j=1}^{i-1}$ *exists with* $K - (a_1v_1 + a_2v_2 + \cdots + a_{i-1}v_{i-1}) \in [\beta, \gamma]$.

Proposition 3 ([1]) *Let ν_1 and ν_2 be nodes of an ROBDD with the same selector variable. If the intervals of ν_1 and ν_2 overlap then $\nu_1 = \nu_2$.*

Proposition 4 *Let $[\beta, \gamma]$ be the interval of a node of an ROBDD. Then $\gamma \geq -1$.*

Proof Suppose $\gamma < -1$. Since $M \leq \gamma < -1$ the node is equivalent to the false node, which has the interval $(-\infty, -1]$. □

3. BDD size of a binary expanded PB constraint

In [1], Abío et al. have shown that for a PB constraint where all coefficients are powers of two the ascending order yields a polynomial sized ROBDD. Here we prove that this also holds for the descending order. Note that this result is a special case of Subsection 4.2.

In this section, we consider a PB constraint C of the following form:

$$\begin{array}{l} (\delta_{0,1} \cdot 2^0)x_{0,1} + \cdots + (\delta_{0,n} \cdot 2^0)x_{0,n} \\ + (\delta_{1,1} \cdot 2^1)x_{1,1} + \cdots + (\delta_{1,n} \cdot 2^1)x_{1,n} \\ + \qquad \cdots \\ + (\delta_{m,1} \cdot 2^m)x_{m,1} + \cdots + (\delta_{m,n} \cdot 2^m)x_{m,n} \leq K, \end{array}$$

where $\delta_{i,r} \in \{0,1\}$ for all i and r. We consider ROBDDs with descending order $x_{0,1} > x_{0,2} > \cdots > x_{0,n} > x_{1,1} > \cdots > x_{m,n}$ in this section.

Lemma 5 *Let $[\beta, \gamma]$ be the interval of a node with selector variable $x_{i,r}$. Then $\beta < (n+r)2^i$.*

Proof Using Proposition 2(i), there must be an assignment to the variables $\{x_{0,1}, \ldots, x_{i,r}\}$ such that

$$\begin{array}{rcl} \beta & = & (\delta_{0,1} \cdot 2^0)x_{0,1} + (\delta_{0,2} \cdot 2^0)x_{0,2} + \cdots + (\delta_{i,r} \cdot 2^i)x_{i,r} \\ & \leq & (\delta_{0,1}2^0 + \cdots + \delta_{0,n}2^0) + \cdots \\ & & + (\delta_{i-1,1}2^{i-1} + \cdots + \delta_{i-1,n}2^{i-1}) \\ & & + (\delta_{i,1}2^i + \cdots + \delta_{i,r}2^i) \\ & \leq & n2^0 + \cdots + n2^{i-1} + r2^i. \end{array}$$

Here $2^0 + 2^1 + \cdots + 2^{i-1} = 2^i - 1$. Thus, $\beta < n2^i + r2^i = (n+r)2^i$. □

Corollary 6 *The number of nodes with selector variable $x_{i,r}$ is bounded by $n + r + 2$. In particular, the size of the ROBDD belongs to $O(n^2 m)$.*

Proof Let $\nu_1, \nu_2, \ldots, \nu_t$ be all the nodes with selector variable $x_{i,r}$. Let $[\beta_j, \gamma_j]$ be the interval of ν_j. From Proposition 3 we can assume, without loss of generality, that $\beta_1 < \beta_2 < \cdots < \beta_t$. Then $-1 \leq \gamma_1 < \beta_2 < \cdots < \beta_t$ by Proposition 4. Due to Proposition 2(ii), there is an assignment such that $K_j := K - ((\delta_{m,n} \cdot 2^m)v_{m,n} + \cdots + (\delta_{i,r+1} \cdot 2^i)v_{i,r+1}) \in [\beta_j, \gamma_j]$. Clearly $K_1 < K_2 < \cdots < K_t$. Hence $K_{j+1} - K_j \geq 2^i$. Since $-1 \leq \gamma_1 < \beta_2 \leq K_2$ using Lemma 5, it holds that $0 \leq K_2$. Combining $\beta_t > K_{t-1} > K_{t-2} + 2^i \geq K_2 + (t-3)2^i \geq (t-3)2^i$ with Lemma 5, we get $(n+r)2^i > \beta_t > (t-3)2^i$ and hence $n + r + 2 \geq t$. □

4. BDD size of a mixed-radix base expanded PB constraint

A *mixed-radix base* is a sequence $\langle b_1, \ldots, b_m \rangle$ of natural numbers and used as a base coding of a number by a sequence of small numbers. For example, time and day uses $\langle 60, 60, 24 \rangle$, where the first number represents seconds in a minute, the second one the minutes in an hour, and the last is for the hours in a day. By using this base, 3610 seconds are coded as 10 seconds, 0 minutes, and 1 hour.

Let $\langle b_1, b_2, \ldots, b_m \rangle$ be a mixed-radix base. We use B_i for the product $b_1 b_2 \cdots b_i$ for $0 \leq i \leq m$. Note that $B_0 = 1$. Using this notation, a sequence $\delta_0, \delta_1, \ldots, \delta_m$, which satisfies that $0 \leq \delta_i < b_{i+1}$ for all $0 \leq i < m$, represents a number $\delta_0 B_0 + \delta_1 B_1 + \cdots + \delta_m B_m$.

Example 7 *Let $\langle b_1, b_2, b_3, b_4 \rangle = \langle 3, 5, 2, 2 \rangle$ be a mixed-radix base. Then $B_0 = 1, B_1 = 3, B_2 = 15, B_3 = 30, B_4 = 60$, and 54 is represented as the sequence $0, 3, 1, 1, 0$ with this base, because $0 \cdot 1 + 3 \cdot 3 + 1 \cdot 15 + 1 \cdot 30 + 0 \cdot 60 = 54$.*

Throughout this section, we consider a base $\langle b_1, \ldots, b_m \rangle$ and a PB constraint C' of the following form:

$$\begin{array}{l} (\delta_{0,1} \cdot B_0)x_{0,1} + \cdots + (\delta_{0,n} \cdot B_0)x_{0,n} \\ + (\delta_{1,1} \cdot B_1)x_{1,1} + \cdots + (\delta_{1,n} \cdot B_1)x_{1,n} \\ + \qquad \cdots \\ + (\delta_{m,1} \cdot B_m)x_{m,1} + \cdots + (\delta_{m,n} \cdot B_m)x_{m,n} \leq K, \end{array}$$

where $0 \leq \delta_{i,r} \leq b_{i+1}$ for all i and r. For the simplicity of the proofs, we assume that $\delta_{m+1,1} = \cdots = \delta_{m+1,n} = 0$ and $b_{m+1} = 1 + \max\{\delta_{m,1}, \ldots, \delta_{m,n}\}$.

4.1 BDD size with an ascending order

In this section we consider an ROBDD of C' with ascending order $x_{0,1} < x_{0,2} < \cdots < x_{0,n} < x_{1,1} < \cdots < x_{m,n}$. We use $b^{\max}$ for the maximum number of b_i's, i.e., $b^{\max} = \max\{b_1, \ldots, b_{m+1}\}$.

Lemma 8 *For all $i \leq m$, let $[\beta, \gamma]$ be the interval of a node with selector variable $x_{i,r}$. Then*

(i) B_i divides β,

(ii) $\beta \leq K$, and

(iii) $K - ((r-1)b^{max} + (n-r+1))B_i < \gamma$.

Proof By Proposition 2(i), β can be expressed as a sum of coefficients all of which are multiples of B_i, thus B_i divides β. Proposition 2(ii) gives an assignment to the variables $\{x_{0,1}, \ldots, x_{i,r}\}$ such that $M \in [\beta, \gamma]$ where

$$M := K - ((\delta_{0,1} \cdot B_0)v_{0,1} + \cdots + (\delta_{i,r-1} \cdot B_i)v_{i,r-1}).$$

Thus $\beta \leq M \leq K - (0 + \cdots + 0) \leq K$, and

$$\begin{array}{rcl} \gamma & \geq & M \\ & \geq & K - ((\delta_{0,1}B_0 + \cdots + \delta_{0,n}B_0) + \cdots \\ & & \quad + (\delta_{i-1,1}B_{i-1} + \cdots + \delta_{i-1,n}B_{i-1}) \\ & & \quad + (\delta_{i,1}B_i + \cdots + \delta_{i,r-1}B_i)) \end{array}$$

Here $\delta_{i,j} \leq b_{i+1} - 1$ for any i, j. We have also $(b_1 - 1) + (b_2 - 1)b_1 + \cdots + (b_i - 1)(b_{i-1} \cdots b_1) = (b_i \cdots b_1) - 1$, i.e., $(b_1 - 1)B_0 + (b_2 - 1)B_1 + \cdots + (b_i - 1)B_{i-1} < B_i$. Thus,

$$\begin{aligned}\gamma &\geq K - (n(b_1 - 1)B_0 + \cdots + n(b_i - 1)B_{i-1} \\ &\qquad + (r-1)(b_{i+1} - 1)B_i) \\ &> K - (nB_i + (r-1)(b_{i+1} - 1)B_i) \\ &= K - ((r-1)b_{i+1} + (n + r - 1))B_i \\ &\geq K - ((r-1)b^{max} + (n + r - 1))B_i. \qquad \square\end{aligned}$$

Corollary 9 *The number of nodes with a selector variable* $x_{i,r}$ *is bounded by* $(r-1)b^{max} - n + r$. *In particular, the size of the ROBDD belongs to* $O(n^2 m)$.

Proof Let $\nu_1, \nu_2, \ldots, \nu_t$ be all the nodes with the selector variable $x_{i,r}$. Let $[\beta_j, \gamma_j]$ be the interval of ν_j for $1 \leq j \leq t$. Since intervals are pairwise disjoint (Proposition 3), we have $\beta_1 < \beta_2 < \cdots < \beta_t$. By Lemma 8(i), we get $\beta_j - \beta_{j-1} \geq B_i$ and in particular $\beta_2 \leq \beta_t - (t-2)B_i$. Combining this with Lemma 8(ii) and (iii), we get $K - ((r-1)b^{max} + (n + r - 1))B_i < \gamma_1 \leq \beta_2 \leq \beta_t - (t-2)B_i \leq K - (t-2)B_i$. Hence $K - ((r-1)b^{max} + (n + r - 1))B_i < K - (t-2)B_i$, i.e., $((r-1)b^{max} + (n + r - 1))B_i < (t-2)B_i$ and hence $(r-1)b^{max} + (n+r-1) > t-2$ which gives $(r-1)b^{max} + n + r \geq t$. □

4.2 BDD size with a descending order

In this section we consider an ROBDD of C' with descending order $x_{0,1} > x_{0,2} > \cdots > x_{0,n} > x_{1,1} > \cdots > x_{m,n}$.

Lemma 10 *Let* $[\beta, \gamma]$ *be the interval of a node with a selector variable* $x_{i,r}$. *Then* $\beta < (n + r(b^{max} - 1))B_i$.

Proof Using Proposition 2(i), there must be an assignment to the variables $\{x_{0,1}, \ldots, x_{i,r}\}$ such that

$$\begin{aligned}\beta &= (\delta_{0,1} \cdot B_0)x_{0,1} + (\delta_{0,2} \cdot B_0)x_{0,2} + \cdots + (\delta_{i,r} \cdot B_i)x_{i,r} \\ &\leq (\delta_{0,1}B_0 + \cdots + \delta_{0,n}B_0) + \cdots \\ &\quad + (\delta_{i-1,1}B_{i-1} + \cdots + \delta_{i-1,n}B_{i-1}) \\ &\quad + (\delta_{i,1}B_i + \cdots + \delta_{i,r}B_i).\end{aligned}$$

Here $\delta_{i,j} \leq b_{i+1} - 1$ for any i, j, and furthermore also $(b_1 - 1)B_0 + (b_2 - 1)B_1 + \cdots + (b_i - 1)B_{i-1} < B_i$. Thus,

$$\begin{aligned}\beta &\leq n(b_1 - 1)B_0 + \cdots + n(b_i - 1)B_{i-1} + r(b_{i+1} - 1)B_i \\ &< nB_i + r(b_{i+1} - 1)B_i \\ &\leq nB_i + r(b^{\max} - 1)B_i. \qquad \square\end{aligned}$$

Corollary 11 *The number of nodes with selector variables* $x_{i,r}$ *is bounded by* $n + r(b^{max} - 1) + 2$. *In particular, the size of the ROBDD belongs to* $O(n^2 m)$.

Proof Let $\nu_1, \nu_2, \ldots, \nu_t$ be all the nodes with selector variable $x_{i,r}$. Let $[\beta_j, \gamma_j]$ be the interval of ν_j. From Proposition 3 we can assume, without loss of generality, that $\beta_1 < \beta_2 < \cdots < \beta_t$. Then $-1 \leq \gamma_1 < \beta_2 < \cdots < \beta_t$ by Proposition 4. Due to Proposition 2(ii), there is an assignment such that $K_j := K - ((\delta_{m,n} \cdot B_m)v_{m,n} + \cdots + (\delta_{i,r+1} \cdot B_i)v_{i,r+1}) \in [\beta_j, \gamma_j]$. Clearly $K_1 < K_2 < \cdots < K_t$. Hence $K_{j+1} - K_j \geq B_i$. Since $-1 \leq \gamma_1 < \beta_2 \leq K_2$ using Lemma 10, it holds that $0 \leq K_2$. Combining $\beta_t > K_{t-1} > K_{t-2} + B_i \geq K_2 + (t-3)B_i \geq (t-3)B_i$ with Lemma 10, we get $(n + r(b^{max} - 1))B_i > \beta_t > (t-3)B_i$ and hence $n + r(b^{max} - 1) + 2 \geq t$. □

Table 1: Number of solved problems

Expan./Order	DEC	OPT		total
	SMALL	BIG	SMALL	
binary/ascending	66	19	78	163
binary/descending	66	19	77	162
mixed/ascending	66	22	93	181
mixed/descending	66	23	75	164
raw/ascending	67	36	108	211
raw/descending	66	25	92	183
MiniSat+	64	20	81	165
MiniSat+ (BDD-only)	67	31	102	200

5. Implementation and experiments

We implemented our findings on top of Minisat+ [5] version 1.0, resulting in the tool GPW. The major extensions are summarized as follows:

- Minisat+ has a function to generate clauses via BDDs constructed from each PB constraint. Thus we attached intervals to the nodes of BDDs to reduce redundant nodes.
- Binary/mixed-radix base expansion of coefficients before BDD construction. We use the optimal-base [4] as a mixed-radix base for each constraint. Currently, we use the function in Minisat+ for sorting networks that minimizes the sum of digits in the expanded constraint, where prime numbers up to 17 are allowed for the radices.

We performed experiments on a machine equipped with dual Xeon W5590 (3.33GHz, 4core 8thread, L2cache4*256KB, and L3cache 8MB) processor and 48GB memory. We used MiniSat version 1.14 as underlying solver. The PB benchmarks consist of 306 problems in total; 81, 80, and 145 problems in DEC-SMALLINT-LIN, OPT-BIGINT-LIN, and OPT-SMALLINT-LIN divisions of Pseudo-Boolean Competition 2010, respectively.

Table 1 shows the number of problems that different methods could solve within 600 seconds timeout. The columns correspond to the divisions of problems. The first six rows show the number of problems solved by GPW where we pre-processed the PB constraints to binary or mixed-radix base (first four rows) or did not pre-process (rows five and six). The last two rows show the results for MiniSat+, where the former uses the default strategy and the latter the "BDD-only" strategy.

Figure 2 shows the total number of solved problems within the timeout for different methods.

Ignoring divisions of problems, raw/ascending (no pre-processing and ascending order) scores best. Ascending

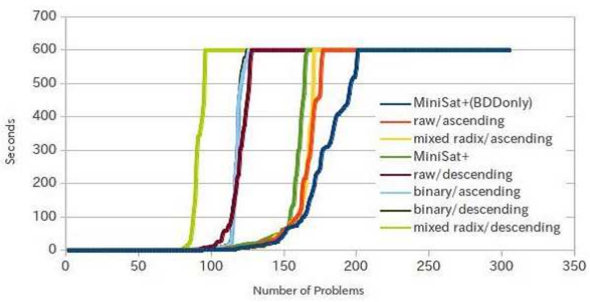

Figure 2: Runtime for the solved problems

order is better than descending. Compared with mixed-radix/ascending and raw/ascending, the former method is faster more than 10 seconds for 17 problems, none of which are BIGINT problems. The latter is faster more than 10 seconds for 61 problems, 17 of which are BIGINT problems. No problems are solved by only the former method, and 29 problems are solved by only the latter method.

All problems solved by binary/ascending are also solved by mixed-radix/ascending. On the other hand, there are 8 problems solved by binary/descending but not solved by mixed-radix/descending.

6. Concluding remarks

We have shown that the ROBDD for a PB constraint whose coefficients are powers of two has polynomial size, and this also holds in the case that each coefficient is expanded by mixed-radix base.

Although the descending order without expansion is essentially the same strategy as MiniSat+, the current implementation works better than MiniSat+ with BDD-only strategy. Possible reasons of the difference are as follows:

- The current implementation of intervals requires constraints to be of the shape $a_1x_1 + \cdots + a_nx_n \leq K$. On the other hand, MiniSat+ constructs a BDD from $K' \leq a_1x_1 + \cdots + a_nx_n \leq K$.
- The current implementation does not allow to mix between non-, binary, or mixed-radix base expansion for each constraint in a PB problem.

The current implementation does not allow to mix between non-, binary, or mixed-radix base expansion for each con- straint in a PB problem. Thus, allowing this may improve the performance. We plan to tackle these issues as future work.

Acknowledgments

Supported by the Austrian Science Fund (FWF) project I963 and the Japan Society for the Promotion of Science.

References

[1] Ignasi Abío, Robert Nieuwenhuis, Albert Oliveras, Enric Rodríguez-Carbonell, and Valentin Mayer-Eichberger. A new look at BDDs for Pseudo-Boolean constraints. *Journal of Artificial Intelligence Research (JAIR)*, 45:443–480, 2012.

[2] Olivier Bailleux, Yacine Boufkhad, and Olivier Roussel. A translation of Pseudo Boolean constraints to SAT. *Journal on Satisfiability, Boolean Modeling and Computation (JSAT)*, 2(1-4):191–200, 2006.

[3] Randal E. Bryant. Graph-based algorithms for boolean function manipulation. *IEEE Trans. Comput.*, 35(8):677–691, August 1986.

[4] Michael Codish, Yoav Fekete, Carsten Fuhs, and Peter Schneider-Kamp. Optimal base encodings for Pseudo-Boolean constraints. In Parosh Aziz Abdulla and K. Rustan M. Leino, editors, *TACAS*, volume 6605 of *Lecture Notes in Computer Science*, pages 189–204. Springer, 2011.

[5] N. Eén and N. Sörensson. Translating Pseudo-Boolean constraints into SAT. *Journal on Satisfiability, Boolean Modeling and Computation (JSAT)*, 2(1–4):1–26, 2006.

[6] Christian Herde. Extending DPLL for Pseudo-Boolean constraints. In *Efficient Solving of Large Arithmetic Constraint Systems with Complex Boolean Structure*, pages 37–58. Vieweg+Teubner, 2011.