New Compilation Languages Based on Restricted Weak Decomposability

Petr Illner

Department of Theoretical Computer Science and Mathematical Logic, Faculty of Mathematics and Physics, Charles University, Czech Republic

illner3@gmail.com

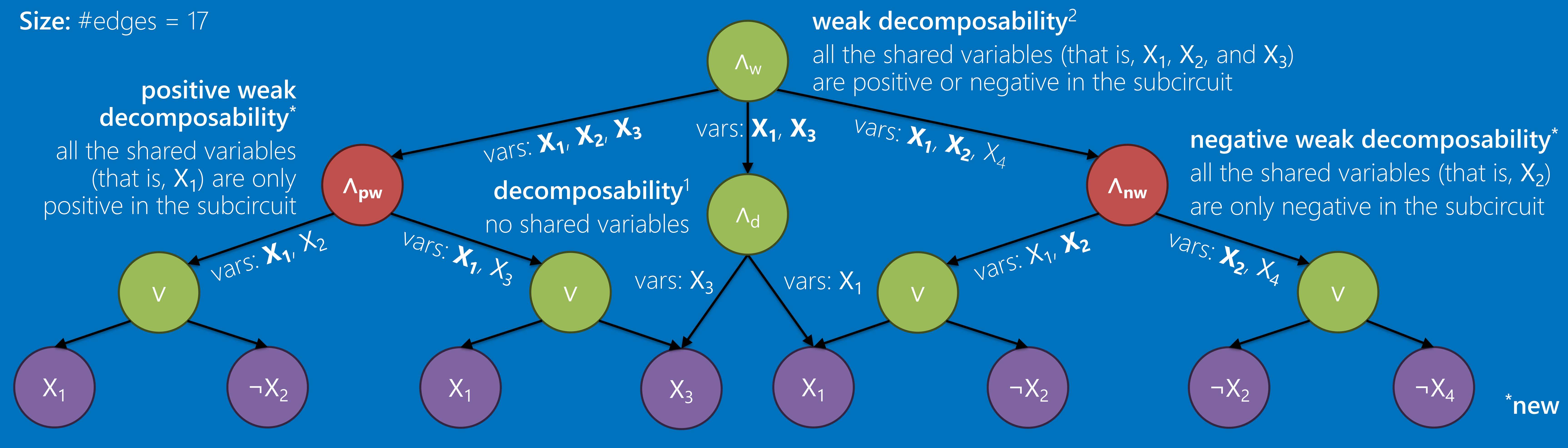
Positive and negative weak decomposable negation normal form (pwDNNF/nwDNNF) circuits

DNNF circuits[1] satisfy the decomposability property

wDNNF circuits[2] satisfy the weak decomposability property

pwDNNF circuits* satisfy the positive weak decomposability property

nwDNNF circuits* satisfy the negative weak decomposability property



Cardinalities

The cardinality of a model is the number of variables that are set to True.

The minimum (resp. maximum) cardinality of a circuit/formula is the minimum (resp. maximum) cardinality of all its models.

Example: $(X_1 \vee X_2) \wedge (X_2 \vee X_3) \wedge (\neg X_1 \vee \neg X_3)$

var	weight
X_1	1
X_2	2
X_3	3

models	cardinality	weighted cardinality
$\neg X_1, X_2, \neg X_3$	**1** (min)	**2** (min)
$\neg X_1, X_2, X_3$	**2** (max)	2 + 3 = **5** (max)
$X_1, X_2, \neg X_3$	**2** (max)	1 + 2 = **3**

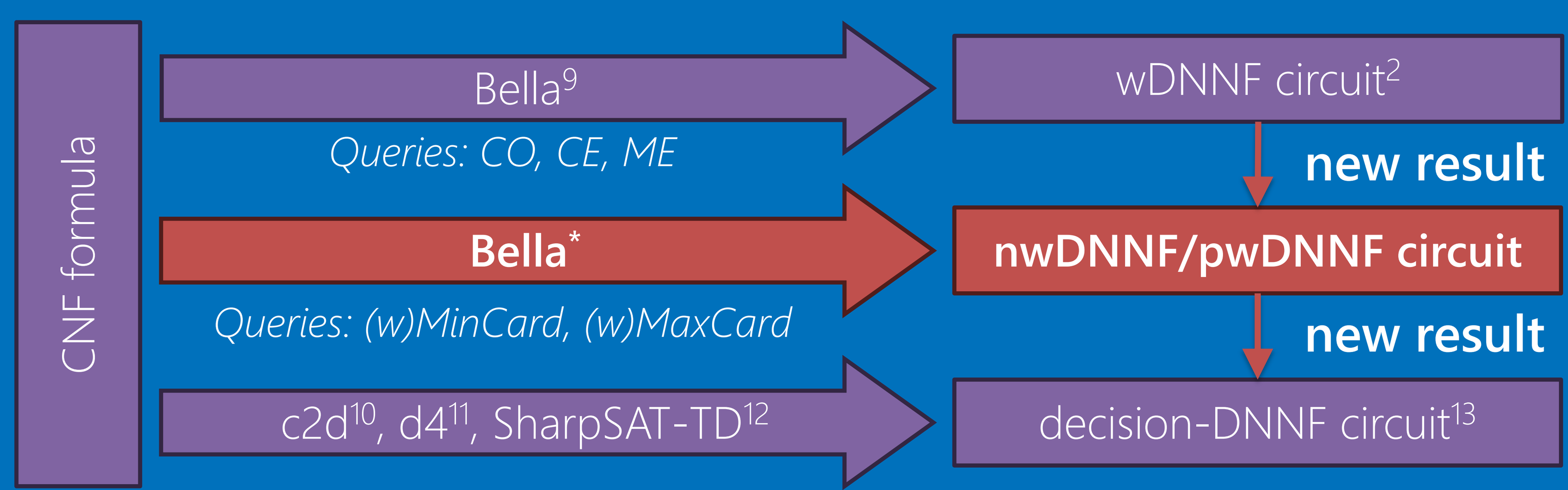
Queries and transformations

Query[3,4,9]	wDNNF	**pwDNNF**	**nwDNNF**	DNNF
Consistency *(CO)*	✓	✓*	✓*	✓
Clausal entailment *(CE)*	✓	✓*	✓*	✓
Model enumeration *(ME)*	✓	✓*	✓*	✓
(weighted) Minimum cardinality *((w)MinCard)*	✗*	✗*	✓*	✓
(weighted) Maximum cardinality *((w)MaxCard)*	✗*	✓*	✗*	✓
Transformation[3,9]	**wDNNF**	**pwDNNF**	**nwDNNF**	**DNNF**
Conditioning *(CD)*	✓	✓*	✓*	✓
Forgetting *(FO)*	✓	✓*	✓*	✓
Disjunction *(∨C)*	✓	✓*	✓*	✓

✓ polytime ✓ polynomial delay ✗ not polytime unless P = NP *new

Knowledge compilers and succinctness[8]

Currently, when we are interested in the (weighted) minimum or maximum cardinality queries, which are essential for many applications, we must use decision-DNNF circuits, which are strictly less succinct than DNNF circuits, which are strictly less succinct than pwDNNF and nwDNNF circuits. Considering those queries, we can use pwDNNF and nwDNNF circuits instead and **our extended variant of Bella – the state-of-the-art compiler for wDNNF circuits.**



L_1 -> L_2 means that L_1 is strictly more succinct than L_2

***our extended variant of Bella[9] for pwDNNF and nwDNNF circuits**

[1] DARWICHE, Adnan. Compiling knowledge into decomposable negation normal form. In: IJCAI. 1999. p. 284-289.
[2] AKSHAY, S., et al. Knowledge compilation for Boolean functional synthesis. In: 2019 Formal Methods in Computer Aided Design (FMCAD). IEEE, 2019. p. 161-169.
[3] DARWICHE, Adnan; MARQUIS, Pierre. A knowledge compilation map. Journal of Artificial Intelligence Research, 2002, 17: 229-264.
[4] DARWICHE, Adnan. Decomposable negation normal form. Journal of the ACM (JACM), 2001, 48.4: 608-647.
[5] PARK, James D. Using weighted MAX-SAT engines to solve MPE. In: AAAI/IAAI. 2002. p. 682-687.
[6] PIPATSRISAWAT, Knot; DARWICHE, Adnan. Clone: Solving weighted Max-SAT in a reduced search space. In: AI 2007: Advances in Artificial Intelligence: 20th Australian Joint Conference on Artificial Intelligence, Gold Coast, Australia, December 2-6, 2007. Proceedings 20. Springer Berlin Heidelberg, 2007. p. 223-233.
[7] LI, Xiao Yu. Optimization algorithms for the minimum-cost satisfiability problem. North Carolina State University, 2004.
[8] GOGIC, Goran, et al. The comparative linguistics of knowledge representation. In: IJCAI (1). 1995. p. 862-869.
[9] ILLNER, Petr; KUČERA, Petr. A Compiler for Weak Decomposable Negation Normal Form. In: Proceedings of the AAAI Conference on Artificial Intelligence. 2024. p. 10562-10570.
[10] DARWICHE, Adnan, et al. New advances in compiling CNF to decomposable negation normal form. In: Proc. of ECAI. Citeseer, 2004. p. 328-332.
[11] LAGNIEZ, Jean-Marie; MARQUIS, Pierre. An Improved Decision-DNNF Compiler. In: IJCAI. 2017. p. 667-673.
[12] KIESEL, Rafael; EITER, Thomas. Knowledge compilation and more with SharpSAT-TD. In: Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning. 2023.
[13] HUANG, Jinbo; DARWICHE, Adnan. The language of search. Journal of Artificial Intelligence Research, 2007, 29: 191-219.

Applications

The most probable explanation (MPE) problem

Instance:	Given a Bayesian network (BN) and some evidence.
Task:	Find a variable instantiation of the remaining variables with the highest probability given that evidence.

Problem: **Computing MPEs in two-layer BNs with large domains**

Required operations: Conditioning, and weighted minimum cardinality

BN → reduction[5,6] → MinCostSat[7] → **Bella** → **nwDNNF circuit**

Experimental results

#nodes	density	domain size	nwDNNF circuits (Bella) time (s)	nwDNNF circuits (Bella) size	nwDNNF circuits (Bella) #	dec-DNNF circuits (D4) time (s)	dec-DNNF circuits (D4) size	dec-DNNF circuits (D4) #
5 : 5	100%	7	**108**	**1 909 132**	**1**	590	6 770 407	1
		8	**459**	**4 209 175**	**1**	3 296	15 836 293	1
		9	**1 884**	**8 471 321**	**1**	---	---	0
	80%	13	**664**	**19 509 851**	**10**	6 625	34 465 982	1
		14	**1 492**	**32 614 471**	**10**	---	---	0
		15	**3 255**	**49 867 671**	**10**	---	---	0
	60%	20	**856**	**29 159 870**	**10**	3 255	26 635 851	8
		25	**2 027**	**34 843 482**	**10**	---	---	0
		28	**3 796**	**52 154 815**	**10**	---	---	0