Smoothing Structured Decomposable Circuits

Andy Shih[1] Guy Van den Broeck[2] Paul Beame[3] Antoine Amarilli[4]

[1]Stanford University
[2]University of California, Los Angeles
[3]University of Washington
[4]LTCI, Télécom Paris, IP Paris

Dec 2019

Arithmetic Circuits

Tractable representation of probabilistic circuits

SOTA for:

- Inference algorithms for PGMs
- Inference algorithms for probabilistic programs
- Discrete density estimation

Check out:
Tractable Probabilistic Models: (UAI19 / AAAI20 tutorial)

Arithmetic Circuits

Can be thought of as a computation graph.

Why do we get tractability?

- Operations are sums and products $\oplus, \otimes$.
- Global / local structures of the computation graph are enforced.
  - Decomposability
  - Determinism
  - Smoothness

Tractability

Different combination of properties leads to different families of ACs

	AC	SPN	PSDD
Decomposability	✓	✓	✓(S)
Determinsim	✓	✗	✓
Smoothness	✓	✓	✓
Pr(evid)	✓	✓	✓
Marginal	✓	✓	✓
MPE	✓	✗	✓
Marginal MAP	✗	✗	✓*
Expectation	✗	✗	✓*

...with different tractability properties.

Expressiveness

Source: Tractable Probabilistic Models tutorial

How expressive are probabilistic circuits?

density estimation benchmarks

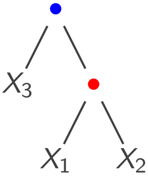
dataset	best circuit	BN	MADE	VAE	dataset	best circuit	BN	MADE	VAE
nltcs	**-5.99**	-6.02	-6.04	**-5.99**	***dna***	**-79.88**	-80.65	-82.77	-94.56
msnbc	**-6.04**	**-6.04**	-6.06	-6.09	***kosarek***	**-10.52**	-10.83	-	-10.64
kdd	-2.12	-2.19	**-2.07**	-2.12	***msweb***	-9.62	-9.70	**-9.59**	-9.73
plants	**-11.84**	-12.65	-12.32	-12.34	***book***	-33.82	-36.41	-33.95	**-33.19**
audio	-39.39	-40.50	-38.95	**-38.67**	***movie***	-50.34	-54.37	-48.7	**-47.43**
jester	-51.29	**-51.07**	-52.23	-51.54	***webkb***	-149.20	-157.43	-149.59	**-146.9**
netflix	-55.71	-57.02	-55.16	**-54.73**	***cr52***	-81.87	-87.56	-82.80	**-81.33**
accidents	-26.89	**-26.32**	-26.42	-29.11	***c20ng***	-151.02	-158.95	-153.18	**-146.9**
retail	**-10.72**	-10.87	-10.81	-10.83	***bbc***	**-229.21**	-257.86	-242.40	-240.94
pumbs*	-22.15	**-21.72**	-22.3	-25.16	***ad***	-14.00	-18.35	**-13.65**	-18.81

71/147

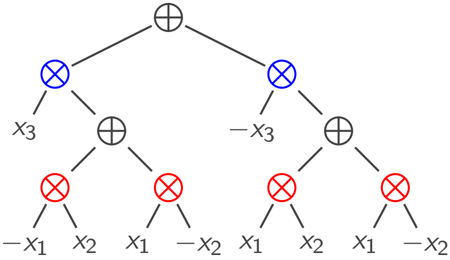
Property 1: Structured Decomposability

Definition

A circuit is **structured decomposable** if each $\otimes$-gate decomposes according to a **vtree**.



(a) vtree v over $X_1X_2X_3$



(b) circuit decomposes according to v

Property 2: Determinism (aka Selectivity)

Definition

A circuit on variables $\mathbf{X}$ is **deterministic** if under any input $\mathbf{x}$, at most one child of each $\oplus$-gate is nonzero.

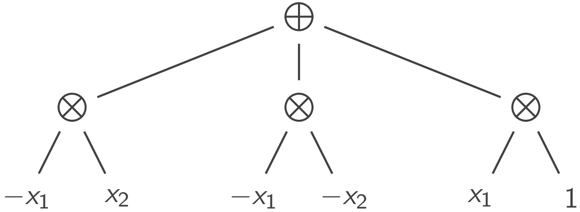
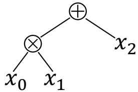


Figure: deterministic circuit over $X_1 X_2$.

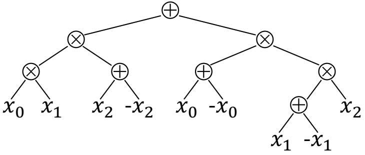
Property 3: Smoothness (aka Completeness)

Definition

A circuit is **smooth** if for every pair of children c_1 and c_2 of a $\oplus$-gate, $vars_{c_1} = vars_{c_2}$.



(a) A circuit.



(b) A smooth circuit.

Figure: Two equivalent circuits computing $(x_0 \otimes x_1) \oplus x_2$. The left one is not smooth and the right one is smooth.

Enforcing Smoothness: Naive Quadratic Algorithm

- Go to each $\oplus$ gate with children $c_1, \dots, c_k$
- Fill in missing variables by attaching $\bigotimes_{\mathbf{x} \in \mathbf{Y}} (\mathbf{x} \oplus -\mathbf{x})$, where $\mathbf{Y} \subseteq \mathbf{X}$
- Complexity $O(nm)$

Problematic when $n \geq 1,000$ and $m \geq 1,000,000$

Enforcing Smoothness: Missing Intervals

Key Insight: for structured decomposable circuits, the missing variables for each gate form two intervals (in the inorder traversal of vtree).

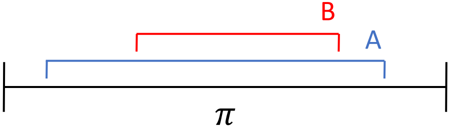


Figure: $A \setminus B$ forms two intervals

We need to fill in $2m$ intervals. Can we try to precompute subintervals and combine them in a smart way?

We can "merge" but we cannot "subtract/split" intervals.

Semigroup Range Sum

Theorem

Given n variables defined over a semigroup and m intervals, the sum of all intervals can be computed using $O(m \cdot \alpha(m, n))$ additions [Chazelle and Rosenberg 1989].

$\alpha(m, n)$ is the inverse Ackermann function, which grows very slowly.

*The original theorem only bounds the number of additions. We bound the number of total operations.

Semigroup Range Sum: Near-Linear Algorithm

Double recursion.

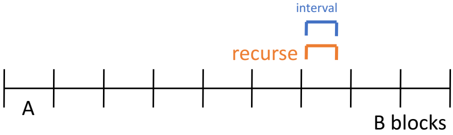


Figure: Inner recursion

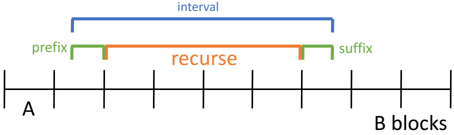
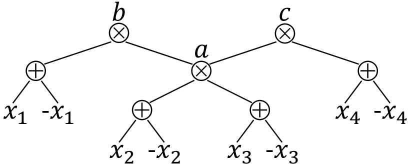


Figure: Outer recursion

Trace additions into circuit

```
a = w(x_2) + w(x_3)
b = a + w(x_1)
c = a + w(x_4)
output b, c
```

(a) Seq of additions to compute intervals



(b) Tracing the additions into a circuit

Figure: Constructing smoothing gates for $\{x_1, x_2, x_3\}$ and $\{x_2, x_3, x_4\}$. Tracing is done by replacing $w(x_i)$ with $x_i \oplus -x_i$ and replacing each addition with a $\otimes$-gate.

More Results

Results on structured decomposable circuits: n is the number of variables and m is the size of the circuit

Task	**Operations**	**Complexity**
Smoothing	$\oplus, \otimes$	$O(m \cdot \alpha(m, n))$
Smoothing*	$\oplus, \otimes$	$\Omega(m \cdot \alpha(m, n))^*$
All-Marginal	$\oplus, \ominus, \otimes, \oslash$	$\Theta(m)$

Matching lower bound* and linear All-Marginal algorithm!

*For *smoothing-gate algorithms* on decomposable circuits.

Thanks!

Poster: East Exhibition Hall B+C #182, 10:45AM

Code: https://github.com/AndyShih12/SSDC

Contact: andyshih@cs.stanford.edu

For more:
- Tractable Probabilistic Models Tutorial at AAAI20
- "A Knowledge Compilation Map" [Darwiche and Marquis]