On the Complexity of Sum-Of-Products Problems over Semirings

Thomas Eiter & Rafael Kiesel

Vienna University of Technology, Vienna, Austria

{thomas.eiter,rafael.kiesel}@tuwien.ac.at

funded by FWF project W1255-N23

Sum-Of-Products [1]

- finite domain $\mathcal{D}$
- functions $f_i : \mathcal{D}^{j_i} \to R, i = 1, \dots, n$
- compute

$$\sum_{X_1,\dots,X_m \in \mathcal{D}} \prod_{i=1}^{n} f_i(\vec{Y_i}), \qquad (1)$$

where $\vec{Y_i}$ is a vector of variables from $\{X_1, \dots, X_m\}$

- The "sum" and the "product" do not need to be the usual addition and multiplication over the reals, but can be any addition $\oplus$ and multiplication $\otimes$ from a semiring $\mathcal{R} = (R, \oplus, \otimes, e_\oplus, e_\otimes)$.

Semirings

A *semiring* $\mathcal{R} = (R, \oplus, \otimes, e_\oplus, e_\otimes)$ consists of a nonempty set R equipped with two binary operations $\oplus$ and $\otimes$, called addition and multiplication, s.t.

$$(a \oplus b) \oplus c = a \oplus (b \oplus c) \quad (a \otimes b) \otimes c = a \otimes (b \otimes c)$$
$$e_\oplus \oplus a = a = a \oplus e_\oplus \qquad e_\otimes \otimes a = a = a \otimes e_\otimes$$
$$a \oplus b = b \oplus a$$
$$a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$$
$$(a \oplus b) \otimes c = (a \otimes c) \oplus (b \otimes c)$$
$$e_\oplus \otimes a = e_\oplus = a \otimes e_\oplus$$

A semiring is *commutative*, if $(R, \otimes)$ is commutative, and is *idempotent*, if $\forall r \in R : r \oplus r = r$.

Well-known Semirings

Some examples of well-known semirings are

- $\mathbb{F} = (\mathbb{F}, +, \cdot, 0, 1)$, for $\mathbb{F} \in \{\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}\}$ the semiring of the numbers in $\mathbb{F}$ with addition and multiplication,
- $\mathcal{R}_{\max} = (\mathbb{N} \cup \{-\infty\}, \max, +, -\infty, 0)$,, the max-plus (max-tropical) semiring,
- $\mathcal{R}_{\min} = (\mathbb{N} \cup \{\infty\}, \min, +, \infty, 0)$, the min-plus (min-tropical) semiring,
- $\mathbb{B} = (\{0, 1\}, \vee, \wedge, 0, 1)$, the Boolean semiring,
- $\mathcal{R}[(x_i)_\alpha] = (R[(x_i)_\alpha], \oplus, \otimes, e_\oplus, e_\otimes)$, for $\alpha \in \mathbb{N}$ (resp. $\alpha = \infty$), is the semiring of polynomials with variables $x_1, \dots, x_\alpha$ (resp. $x_1, x_2, \dots$) over the semiring $\mathcal{R}$.

Motivation

For some semirings the associated Sum-Of-Products problem and the complexity thereof is well-known:

Problem	Instance	Semiring	Complexity
▷ SAT	$\bigvee_{a_1,\dots,a_n \in \{0,1\}} \bigwedge_{j=1}^{m} C_j$	$\mathbb{B}$	NP-comp.
▷ WeightedMaxSAT	$\max_{a_1,\dots,a_n \in \{0,1\}} \sum_{j=1}^{m} w_j \mathbb{1}_{C_j}$	$\mathcal{R}_{\max}$	OptP-comp.
▷ #SAT	$\sum_{a_1,\dots,a_n \in \{0,1\}} \prod_{j=1}^{m} \mathbb{1}_{C_j}$	$\mathbb{N}$	#P-comp.

There are more Sum-Of-Products problems that are also relevant for which the complexity has yet to be considered.

Problem	Semiring	Complexity
▷ Most Probable Explanation	$([0,1], \max, \cdot, 0, 1)$	?
▷ Sensitivity Analysis	$(\mathbb{R}_{\geq 0}[\mathcal{V}], +, \cdot, 0, 1)$	?
▷ Gradient Computation	$GRAD$	?
▷ SumProd	$(R, \oplus, \otimes, e_\oplus, e_\otimes)$	?

Apart from instances over fixed semirings, there are also frameworks, whose semantics was parameterized with semirings to allow quantitative reasoning in a general form [2–4].

Semiring Turing Machines

- Aim: Capture SAT($\mathcal{R}$) but not more.
- Allow semiring values $r \in R$ on the tape.
- Use a *weighted* transition relation $\delta \subseteq (Q \times (\Sigma \cup R)) \times (Q \times (\Sigma \cup R)) \times \{-1, 1\} \times R$.
- This is too strong! Need restrictions on δ. For each $((q_1, \sigma_1), (q_2, \sigma_2), e, r) \in \delta$:
  1. *cannot write or overwrite semiring values:* $\sigma_1 \in R$ or $\sigma_2 \in R$ implies $\sigma_1 = \sigma_2$
  2. *transition only with $r \in R'$ or value under head:* $r \in R'$ or $r = \sigma_1 \in R$
  3. *cannot differentiate semiring values:* $\sigma_1 \in R$ implies that for all $\sigma_1' \in R$ we have $((q_1, \sigma_1'), (q_2, \sigma_1'), e, r') \in \delta$, where $r' = \sigma_1'$ if $r = \sigma_1$ and else $r' = r$

Semiring Turing Machine Output

Let M be an SRTM and $c = (q, w, n)$ a configuration, where q is a state, w is the string on the tape and n is the head position. The value $v(c)$ of c w.r.t. M is

- $e_\otimes$, if there are no possible transitions from c to another configuration
- $\bigoplus_{c \xrightarrow{r} c'} r \otimes v(c')$, otherwise, where $c \xrightarrow{r} c'$ denotes that M can transit from c to c' with weight r

The output is $v(c_0)$, the value of the initial configuration c_0.

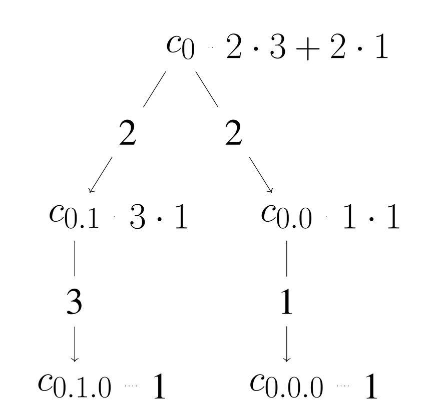

Figure 1: A computation tree over $\mathbb{N}$. Each transition $c \xrightarrow{r} c'$ is annotated with its weight r and each configuration c is annotated with its value $v(c)$.

NP($\mathcal{R}$) is the class of all functions computable in polynomial time by an SRTM over $\mathcal{R}$.

Theorem: NP($\mathcal{R}$)-completeness

SAT($\mathcal{R}$) is NP($\mathcal{R}$)-complete with respect to polynomial transformations, for every semiring $\mathcal{R}$.
Further, the following problems are NP($\mathcal{R}$)-complete by reduction from SAT($\mathcal{R}$):

- Sum-of-Products [1]
- Semiring-based Constraint Satisfaction Problems [2]
- Algebraic Model Counting [4]
- Algebraic Constraint Evaluation [3]

Classical Complexity

This result gives us

- an insight into how Sum-Of-Products problems can be solved independently of how the semiring values are encoded and how addition and multiplication are given.
- a machinery to approach other problems that are parameterized with semirings.

But: How hard is the problem in terms of classical complexity?

Semiring Complexity Map

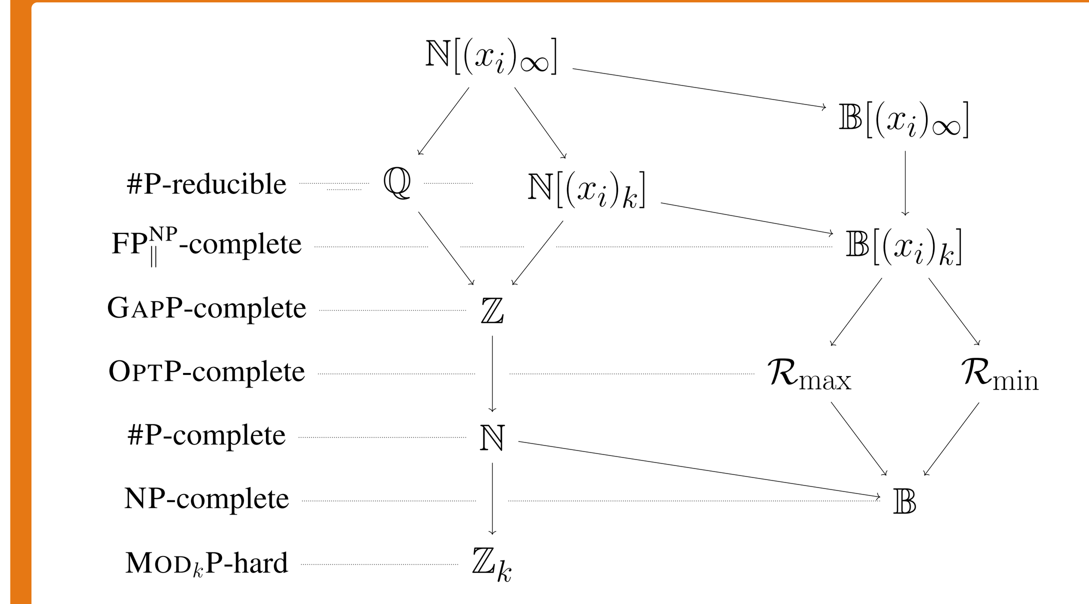

Figure 2: Epimorphisms $f : \mathcal{R}_1 \to \mathcal{R}_2$ between semirings, indicated by arrows $\mathcal{R}_1 \to \mathcal{R}_2$. Relation of complexity classes $\mathcal{C}$ and semirings $\mathcal{R}$, indicated by dotted lines $\mathcal{C} \cdots \mathcal{R}$.

Encodings

- Classical model of computation
  ↪ assume semiring values to be encoded by an *injective* function $e : R \to \{0, 1\}^*$, called *encoding (function)*.
- Complexity depends on the encoding:
  - With respect to the binary encoding Knapsack is NP-hard.
  - With respect to the unary encoding Knapsack is in P.
- Even worse, there is a semiring whose multiplication is *undecidable* or *linear time* depending on the encoding.

Sources of Complexity

1. Encoding of the input
2. Information retained by addition and multiplication:
   - $c_1 \vee c_2$ over $\mathbb{B}$ retains whether both c_1, c_2 are 0
   - $c_1 + c_2$ over $\mathbb{N}$ retains the sum of c_1, c_2
   - $c_1 x_1 + c_2 x_2$ over $\mathbb{N}[x_1, x_2]$ retains the values c_1, c_2

We consider 2.

Epimorphisms

Let $\mathcal{R}_i = (R_i, \oplus_i, \otimes_i, e_{\oplus_i}, e_{\otimes_i}), i = 1, 2$ be semiring. Then an *epimorphism* is a surjective function $f : R_1 \to R_2$ such that for $\odot = \oplus, \otimes$

$$f(r \odot_1 r') = f(r) \odot_2 f(r') \text{ and } f(e_{\odot_1}) = e_{\odot_2}.$$

Intuitively, if there is an epimorphism from $\mathcal{R}_1$ to $\mathcal{R}_2$, then $\mathcal{R}_1$ retains at least as much information as $\mathcal{R}_2$. For an example consider Figure 3.

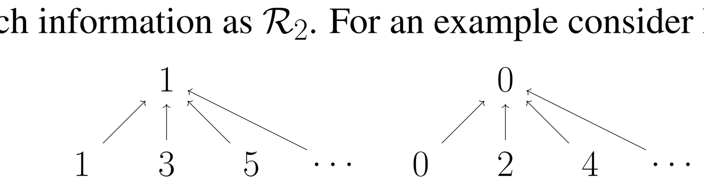

Figure 3: Visualization of the epimorphism between $\mathbb{N}$ and $\mathbb{Z}_2$ that assigns every natural number 0, 1 depending on whether it is even or odd.

Theorem: Epimorphisms are Reductions

Let $e_i(\mathcal{R}_i), i = 1, 2$ be two encoded semirings, such that

1. SAT($e_1(\mathcal{R}_1)$) is in FPSpace(poly),
2. there exists a polynomial time computable epimorphism $f : e_1(R_1) \to e_2(R_2)$, and
3. for each $e_2(r_2) \in e(R_2)$ one can compute in polynomial time $e_1(r_1)$ s.t. $f(e_1(r_1)) = e_2(r_2)$ from $e_2(r_2)$.

Then SAT($e_2(\mathcal{R}_2)$) is counting-reducible to SAT($e_1(\mathcal{R}_1)$).

Applying Epimorphism Reductions

- Approach: find membership results for high information retainers. See Figure 2 for an overview.
- $\mathbb{N}[(x_i)_\infty], \mathbb{B}[(x_i)_\infty]$ have epimorphisms to every commutative countable (resp. and idempotent) semiring! Unfortunately:

Negative Results

Let $\mathcal{R} = \mathbb{N}[(x_i)_\infty]$ (resp. $\mathcal{R} = \mathbb{B}[(x_i)_\infty]$).
The following are equivalent:

1. There is an encoding function e for $\mathcal{R}$ s.t.
   1) $\|[\![\alpha]\!]_{\mathcal{R}}(\mathcal{I})\|_e$ is polynomial in the size of $\alpha, \mathcal{I}$,
   2) we can extract the coefficient of $x_{i_1}^{j_1} \dots x_{i_n}^{j_n}$ from $e(r)$ in polynomial time in $\|r\|_e$, and
   3) $\|x_i\|_e$ is polynomial in i,
2. #P $\subseteq$ FP/poly (resp. NP $\subseteq$ P/poly).

- Link to open complexity theoretic questions!

Finitely Generated Semirings

Unlikely to work in general: consider subclasses!

Positive Results

Let e be the encoding function that represents exponents in unary and coefficients in binary. Then

- SAT($e(\mathbb{Q}[(x_i)_k])$) is counting-reducible to #SAT and #P-hard for counting reductions.
- SAT($e(\mathbb{B}[(x_i)_k])$) is $\mathrm{FP}^{\mathrm{NP}}_{\parallel}$-complete for metric reductions.

- Let $\mathcal{R} = (R, \oplus, \otimes, e_\oplus, e_\otimes)$ be a semiring.
- The *semiring generated* by a subset $S \subseteq R$ is defined as

$$\langle S \rangle_{\mathcal{R}} := \bigcap \{R' \subseteq R \mid S \subseteq R', (R', \oplus, \otimes, e_\oplus, e_\otimes) \text{ is a semiring}\}.$$

- $\mathcal{R}$ is *finitely generated* if $\langle S \rangle_{\mathcal{R}} = R$ for $S = \{r_1, \dots, r_n\}$.
- If $\mathcal{R}$ is finitely generated and commutative, then there is an epimorphism from $\mathbb{N}[(x_i)_n]$ to $\mathcal{R}$. If $\mathcal{R}$ is further idempotent there is even an epimorphism from $\mathbb{B}[(x_i)_n]$.
  ↪ Idea: Use reductions to SAT($\mathbb{N}[(x_i)_n]$) (resp. SAT($\mathbb{B}[(x_i)_n]$))!

Takeaway

- Sum-Of-Products over $\mathcal{R}$ is NP($\mathcal{R}$)-complete
- The encoding matters
- Over general semiring Sum-Of-Products is unlikely to have polynomial outputs
- There are broad classes of countable commutative (resp. and idempotent) semirings s.t. Sum-Of-Products is not much harder than #SAT (resp. SAT)

References

[1] Bacchus, F., Dalmao, S., Pitassi, T.: Solving# sat and bayesian inference with backtracking search. JAIR **34**, 391–442 (2009)

[2] Bistarelli, S., Montanari, U., Rossi, F., Schiex, T., Verfaillie, G., Fargier, H.: Semiring-based csps and valued csps: Frameworks, properties, and comparison. Constraints **4**(3), 199–240 (1999)

[3] Eiter, T., Kiesel, R.: Asp(ac): Answer set programming with algebraic constraints. arXiv preprint arXiv:2008.04008 (2020)

[4] Kimmig, A., Van den Broeck, G., De Raedt, L.: Algebraic model counting. Journal of Applied Logic **22**, 46–62 (2017)