Information Processing Letters ••• (••••) •••–•••

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Hard problems in max-algebra, control theory, hypergraphs and other areas ☆

Marc Bezem [a], Robert Nieuwenhuis [b], Enric Rodríguez-Carbonell [b,*]

[a] Department of Informatics, University of Bergen, Postboks 7800, N-5020 Bergen, Norway
[b] Technical University of Catalonia (UPC), Barcelona, Jordi Girona 1-3, 08034 Barcelona, Spain

ARTICLE INFO

Article history:
Received 16 October 2009
Accepted 18 November 2009
Available online xxxx
Communicated by A.A. Bertossi

Keywords:
Algorithms
Computational complexity
(Hyper)graphs
Constraint solving

1. Results

We introduce the *max-atom problem* (MAP): solving (in $\mathbb{Z}$) systems of inequations of the form $\max(x, y)+k \geqslant z$, where x, y, z are variables and $k \in \mathbb{Z}$. Our initial motivation for MAP was reasoning on delays in circuits using SAT Modulo Theories [10], viewing MAP as a natural extension of Difference Logic, i.e., inequations of the form $x+k \geqslant y$.

Here we show that MAP is PTIME-equivalent to several rather different well-known problems for which no PTIME algorithm has been found so far, in spite of decades of independent efforts. One is on solving *two-sided linear max-plus systems* (Section 3 of this paper) that arise in Control Theory when modeling Discrete Event Systems, and another one on shortest paths in directed weighted hypergraphs (Section 4).

Interestingly (see Section 2), there is also a simple PTIME equivalence between MAP and a scheduling problem considered in [9], namely computing earliest job start times for systems of AND/OR precedence constraints, which is proved in [9] to be in turn PTIME-equivalent to Mean Payoff Games (MPG), a well-known hard problem in NP ∩ Co-NP.

Therefore, it is not so surprising any more that no PTIME algorithms had been found for the aforementioned problems on hypergraphs and Discrete Event Systems.

Note that solving MAP in PTIME would imply the same for Parity Games (via MPG [5]) and hence for model checking in the propositional μ-calculus [6], which is very important in verification. As an example of interesting new insights from MAP,[1] in Section 5 we show that a PTIME algorithm for MAP over $\mathbb{Z}$ also gives a PTIME algorithm over $\mathbb{Q}$, but that a weakly polynomial algorithm we give for $\mathbb{Z}$ does *not* carry over to $\mathbb{Q}$, so, unlike what happens

☆ Partially supported by Spanish Ministry of Education & Science LogicTools-2 project (TIN2007-68093-C02-01).

* Corresponding author.
E-mail address: erodri@lsi.upc.edu (E. Rodríguez-Carbonell).
URLs: http://www.ii.uib.no/~bezem (M. Bezem), http://www.lsi.upc.edu/~roberto (R. Nieuwenhuis), http://www.lsi.upc.edu/~erodri (E. Rodríguez-Carbonell).

[1] Note: a preliminary version of this paper is [3], in which we were not aware yet of [9] and the PTIME-equivalences with MPGs. In [3] we also gave independent simple proofs of membership in NP ∩ Co-NP.

0020-0190/$ – see front matter
doi:10.1016/j.ipl.2009.11.007

Please cite this article in press as: M. Bezem et al., Hard problems in max-algebra, control theory, hypergraphs and other areas, Information Processing Letters (2009), doi:10.1016/j.ipl.2009.11.007

in Linear Programming, in this sense MAP might be harder over $\mathbb{Q}$ than over $\mathbb{Z}$.

2. Simple equivalences with MAP

MAP is quite expressive. Difference logic literals $x+k \geqslant y$ can of course be expressed as $\max(x,x)+k \geqslant y$. Equalities $\max(x,y)+k=z$ can be written as $\max(x,y)+k \geqslant z \wedge z-k \geqslant x \wedge z-k \geqslant y$. Strict inequalities $\max(x,y)+k>z$ can be expressed as $\max(x,y)+k-1 \geqslant z$. One can express max on both sides, as in $\max(x,y)+k=\max(x',y')+k'$ by introducing a fresh variable z and writing $\max(x,y)+k = z \wedge \max(x',y')+k'=z$. One can also express different offsets on different arguments of max; for instance $\max(x+5, y-3) \geqslant z$ can be written as $\max(x,y')+5 \geqslant z \wedge y'+8=y$, where y' is fresh. Furthermore, since $\max(e_1,e_2,e_3)$ is equivalent to $\max(e_1,\max(e_2,e_3))$, one can express nested or larger-arity max-atoms such as $\max(e_1,e_2,e_3) \geqslant z$ by writing $\max(e_1,x) \geqslant z \wedge \max(e_2,e_3)=x$, where x is fresh.

Another less trivial equivalence of MAP is with the problem of deciding the existence of super fixpoints of min–max functions [8]. A *min–max* function is a function $f:\mathbb{Z}^n \to \mathbb{Z}^n$ whose coordinates are min–max expressions, i.e., terms in the grammar $Y \to \min(Y,Y), \max(Y,Y), Y+k, x_1, \ldots, x_n$, where Y is a non-terminal symbol, $k \in \mathbb{Z}$ and $x_1, \ldots, x_n$ are variables. A *super fixpoint* of a min–max function f is $v \in \mathbb{Z}^n$ such that $f(v) \geqslant v$. An instance of MAP can be easily rewritten into the form $f(v) \geqslant v$: one just needs to take for the i-th coordinate of f the minimum of the left-hand sides of the max-atoms in which the variable x_i appears on the right-hand side. Conversely, the problem of determining if a min–max function f admits a super fixpoint can be reduced linearly to MAP, by rewriting $\min(X,Y) \geqslant z$ into $X \geqslant z, Y \geqslant z$, where X, Y are min–max expressions and z is a variable; and $\max(X,Y) \geqslant z$ into $\max(x,y) \geqslant z, X \geqslant x, Y \geqslant y$, where x and y are fresh variables.

A more significant relationship is with the problem of computing earliest job start times for the systems of AND/OR precedence constraints of [9]. To show PTIME-equivalence with MAP, simple syntactic transformations suffice, like interchanging min with max and $\leqslant$ with $\geqslant$. As a consequence, these problems are both in NP ∩ Co-NP (see [9] for details).

3. Equivalence with two-sided linear max-plus systems

Definition 1. *Two-sided linear max-plus systems* are sets of equations of the form $\max(x_1+k_1, \ldots, x_n+k_n) = \max(x_1+k'_1, \ldots, x_n+k'_n)$ where *all* n variables of the system occur on both sides of every equation.

Finding a polynomial algorithm for solving such systems over $\mathbb{Z}$ has been open for more than 30 years in the area of max-plus algebras [4]. An elegant algorithm was given and claimed to be polynomial in [4], but unfortunately in [2] we gave a family of examples on which it behaves exponentially.

Definition 2. Given a set of variables V, the *size* of an assignment $\alpha : V \to \mathbb{Z}$ is the difference between the largest and the smallest value assigned to the variables, i.e., $\text{size}(\alpha) = \max_{x,y \in V}(\alpha(x)-\alpha(y))$.

Lemma 1 *(Small Model Property). If a set of max-atoms S is satisfiable, then it has a model of size at most the sum of the absolute values of the offsets, i.e., at most*

$$K_S = \sum_{\max(x,y)+k \geqslant z \in S} |k|.$$

Proof. We may assume that all constraints are equations: replace each $\max(x,y)+k \geqslant z$ by $\max(x,y)+k=z'$ and $\max(z,z')=z'$. The class of models does not change essentially, and the sum of the absolute values of the offsets is the same. So we assume that S is a set of equations $\max(x,y)+k=z$.

Let α be a model of S. Based on α we define a weighted graph whose vertices are the variables. For every constraint $\max(x,y)+k=z$, if $\alpha(x) \geqslant \alpha(y)$ then we add a red edge (x,z) with weight k and a green edge (y,x) without a weight; and otherwise, if $\alpha(y) > \alpha(x)$ then we add a red edge (y,z) with weight k and a green edge (x,y) without a weight. While changing the model, the graph will remain all the time the same.

A red (weakly) connected component is a subgraph such that there are red paths between any two variables in the subgraph, where the red edges may be used in any direction. The *segment* of a red connected component is the range of integers from the lowest value to the highest one assigned to the variables in the component. The size of such a segment is at most the sum of the absolute values of the weights of the edges in the component.

Red connected components partition variables. If their segments overlap, then $\text{size}(\alpha) \leqslant K_S$. If there is a gap, say of size p, then it is closed by a suitable translation, e.g., by decreasing by p all values assigned to variables above the gap. This respects all red edges and their weights since the gap is between segments of red connected components and components are translated as a whole. Green edges are also respected since we only close gaps and never a variable x with initially a higher value than another variable y ends up with a value strictly lower than y. Since all edges are respected we keep a model, all the time closing gaps until there are no gaps left. We end up with a model α' without gaps and hence $\text{size}(\alpha') \leqslant K_S$. □

Notice that the previous lemma gives a proof of membership of MAP in NP: it suffices to guess a "small" assignment; checking that it is indeed a model is trivially in P.

Theorem 1. *MAP and the problem of satisfiability of a two-sided linear max-plus system are polynomially reducible to each other.*

Proof. Reducing this kind of max-equations to max-atoms can be done as explained in the introduction. For the reverse reduction, by the Small Model Property (Lemma 1), if S is satisfiable then it has a model α such that $\text{size}(\alpha) \leqslant K_S$ (notice that K_S can be computed in polynomial time). Let $V=\{x_1, \ldots, x_n\}$ be the set of variables over which S is

Please cite this article in press as: M. Bezem et al., Hard problems in max-algebra, control theory, hypergraphs and other areas, Information Processing Letters (2009), doi:10.1016/j.ipl.2009.11.007

defined. Now, for each variable x_i, we consider the equation

$$\max(x_1 - 1, \ldots, x_{i-1} - 1, x_i + K_S, x_{i+1} - 1, \ldots, x_n - 1) = \max(x_1, \ldots, x_{i-1}, x_i + K_S, x_{i+1}, \ldots, x_n),$$

which is equivalent to $x_i + K_S \geqslant x_j$, i.e., $K_S \geqslant x_j - x_i$ for all j in $1 \ldots n$, $j \neq i$. Let S'_0 be the two-sided linear max-plus system consisting of these n equations. Now we add new equations to S'_0 to obtain a system S' which is equisatisfiable to S. This is achieved by replacing every max-atom $\max(x_{i_1}, x_{i_2}) + k \geqslant x_{i_3}$ in S by the equation

$$\max\big(x_{i_1} + k, x_{i_2} + k, x_{i_3}, x_j - K_S - |k| - 1, \ldots\big) = \max\big(x_{i_1} + k, x_{i_2} + k, x_{i_3} - 1, x_j - K_S - |k| - 1, \ldots\big),$$

where j ranges over all variable indices different from i_1, i_2, i_3 (if any of the indices i_1, i_2 or i_3 coincide, an obvious simplification must be applied). The offset $-K_S - |k| - 1$ has been chosen so that variables with this offset do not play a role in the maxima. If we leave them out, it is clear that the resulting constraint $\max(x_{i_1} + k, x_{i_2} + k, x_{i_3}) = \max(x_{i_1} + k, x_{i_2} + k, x_{i_3} - 1)$ is equivalent to the max-atom $\max(x_{i_1}, x_{i_2}) + k \geqslant x_{i_3}$. □

4. Equivalence with shortest hyperpaths

In hypergraphs, an edge goes from a set of vertices to another vertex. Hence a natural notion of a hyperpath (from a set of vertices to a vertex) is a tree, and a natural notion of length of the hyperpath is the maximal length (the sum of the weights) of a path from a leaf to the root of this tree (see [1,7]). This is formalized as follows.

A (directed, weighted) *hypergraph* is a tuple $H = (V, E, W)$ where V is the set of *vertices* (here we consider V is finite), E is the set of *hyperedges* and $W : E \to \mathbb{Z}$ is the *weight function*. Each hyperedge is a pair (S, t) from a non-empty finite subset of vertices $S \subseteq V$ called the *source set* to a vertex $t \in V$ called the *target vertex*.

Given a hypergraph $H = (V, E, W)$, a subset of vertices $X \subseteq V$, $X \neq \emptyset$ and $y \in V$, a *hyperpath from X to y* is a tree defined recursively as follows: (i) if $y \in X$, then the empty tree $\emptyset$ is a hyperpath from X to y; (ii) if there is a hyperedge $(Z, y) \in E$ and hyperpaths t_{X,z_i} from X to z_i for each $z_i \in Z$, then the tree $t_{X,y}$ with root (Z, y) and children the trees t_{X,z_i} for each vertex $z_i \in Z$, is a hyperpath from X to y.

The *weight* $\omega(p)$ of a hyperpath p is defined as: (i) if p is $\emptyset$, then $\omega(p) = 0$; (ii) if p is a tree with root the hyperedge e and children $p_1, \ldots, p_m$, then $\omega(p) = W(e) + \max(\omega(p_1), \ldots, \omega(p_m))$.

Given a non-empty subset of vertices $X \subseteq V$, $X \neq \emptyset$, the *distance* function $\delta_X : V \to \mathbb{Z} \cup \{\pm\infty\}$ is defined as $\delta_X(y) = \inf\{\omega(p_{X,y}) \mid p_{X,y}$ is a hyperpath from X to $y\}$. The distance function δ_X is said to be *well-defined* if $\delta_X(y) > -\infty$ for all vertices $y \in V$.

Intuitively, here $+\infty$ means "no hyperpath" and $-\infty$ is related to negative cycles, for instance in the presence of a hyperedge such as $W(\{x\}, x) = -1$. We now show that MAP is PTIME-equivalent to the problem of, given a hypergraph $H = (V, E, W)$, deciding whether δ_X is well-defined for all non-empty $X \subseteq V$.

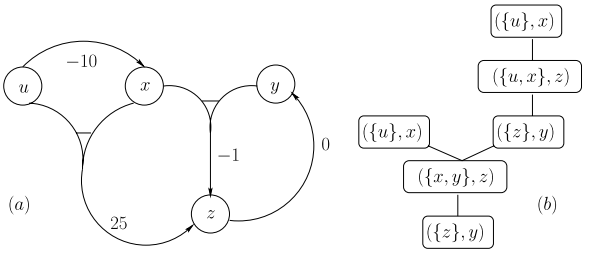

Fig. 1. Example of hypergraph.

Example 1. Fig. 1(a) shows an example of a hypergraph. E.g., the hyperedge $(\{u\}, x)$ has weight -10, while the weight of the hyperedge $(\{u, x\}, z)$ is 25. The empty tree is a hyperpath from $\{u, y\}$ to y with rank 0; Fig. 1(b) shows another hyperpath from $\{u, y\}$ to y, with rank 24.

Lemma 2. *Let $H = (V, E, W)$ be a hypergraph and $X \subseteq V$, $X \neq \emptyset$ be a set of vertices such that $-\infty < \delta_X(y) < +\infty$ for all $y \in V$. If $(Z, y) \in E$, then $\delta_X(y) \leqslant W(Z, y) + \max_{z \in Z}(\delta_X(z))$.*

Proof. By hypothesis for all $y \in V$ we have $-\infty < \delta_X(y) < +\infty$. Thus, in particular, for all $z \in Z$ there exists a hyperpath t_z from X to z such that $\omega(t_z) = \delta_X(z)$. Now the tree t with root (Z, y) and children the trees t_z for each $z \in Z$ is a hyperpath from X to y. So $\delta_X(y) \leqslant \omega(t) = W(Z, y) + \max_{z \in Z}(\omega(t_z)) = W(Z, y) + \max_{z \in Z}(\delta_X(z))$. □

Lemma 3. *Let $H = (V, E, W)$ be a hypergraph and $\alpha : V \to \mathbb{Z}$ be such that $\alpha(y) \leqslant \max_{z \in Z}(\alpha(z)) + W(Z, y)$ for all hyperedges $(Z, y) \in E$. If t is a hyperpath from a non-empty $X \subseteq V$ to $y \in V$, then $\alpha(y) \leqslant \max_{x \in X}(\alpha(x)) + \omega(t)$.*

Proof. Let us prove it by induction over the depth of t. In the base case $t = \emptyset$, and therefore $y \in X$. Since $\omega(\emptyset) = 0$, trivially $\alpha(y) \leqslant \max_{x \in X}(\alpha(x)) = \max_{x \in X}(\alpha(x)) + \omega(\emptyset)$. Now, if t has positive depth, its root is a hyperedge $(Z, y) \in E$, and its children are trees $t_1, \ldots, t_m$ connecting X to $z_1, \ldots, z_m$ respectively, where $Z = \{z_1, \ldots, z_m\}$. By induction hypothesis, for each i in $1 \ldots m$ we have $\alpha(z_i) \leqslant \max_{x \in X}(\alpha(x)) + \omega(t_i)$. Now:

$$\begin{aligned}
\alpha(y) &\leqslant \max_{1 \leqslant i \leqslant n}\big(\alpha(z_i)\big) + W(Z, y) \\
&\leqslant \max_{1 \leqslant i \leqslant n}\Big(\max_{x \in X}\big(\alpha(x)\big) + \omega(t_i)\Big) + W(Z, y) \\
&= \max_{x \in X}\big(\alpha(x)\big) + \max_{1 \leqslant i \leqslant n}\big(\omega(t_i)\big) + W(Z, y) \\
&= \max_{x \in X}\big(\alpha(x)\big) + \omega(t). \qquad \square
\end{aligned}$$

Finally we are in condition to prove the equivalence of the two problems. For convenience, in what remains of this section we assume max-atoms to be of the form $\max_{1 \leqslant i \leqslant n}(x_i) + k \geqslant z$.

Theorem 2. *MAP and the problem of well-definedness of the distance functions of all subsets of vertices of a hypergraph are polynomially reducible to each other.*

Please cite this article in press as: M. Bezem et al., Hard problems in max-algebra, control theory, hypergraphs and other areas, Information Processing Letters (2009), doi:10.1016/j.ipl.2009.11.007

Proof. First we prove that, given a set S of max-atoms, one can compute in polynomial time a hypergraph $H(S)$ whose distance functions are well-defined if and only if S is satisfiable.

Let S be a set of max-atoms over the variables V. We can assume w.l.o.g. that there exists a variable $x \in V$ such that there are max-atoms $x \geqslant y \in S$ for every $y \in V$ (adding a fresh variable with these properties preserves satisfiability). The hypergraph $H(S)$ is defined as follows: its set of vertices is V; and for each max-atom $\max_{z\in Z}(z) + k \geqslant y$, we define a hyperedge $e = (Z, y)$ with weight $W(e) = k$. For example, the hypergraph corresponding to the set of max-atoms $S = \{u - 10 \geqslant x, z \geqslant y, \max(x, y) - 1 \geqslant z, \max(x, u) + 25 \geqslant z\}$ is that shown in Fig. 1(a).

Let us see that the distance function δ_x in $H(S)$ is well-defined if and only if S is satisfiable (we write δ_x instead of $\delta_{\{x\}}$ for the sake of clarity). Let us prove that if δ_x is well-defined then S is satisfiable. By construction, for each max-atom $\max_{z\in Z}(z) + k \geqslant y \in S$ there exists a hyperedge $e = (Z, y)$ in $H(S)$ with weight $W(e) = k$. Now, since δ_x is well-defined and all vertices are hyperconnected to $\{x\}$, by Lemma 2 we have $\max_{z\in Z}(\delta_x(z)) + W(Z, y) \geqslant \delta_x(y)$, and so $\delta_x \models S$. Let us prove the converse, i.e., that if S is satisfiable then δ_x is well-defined, by contradiction. Let us assume that δ_x is not well-defined and let α be a model of S. Then there is $y \in V$ such that $\delta_x(y) = -\infty$. This implies that for all $w \in \mathbb{Z}$ there exists a hyperpath t_w from $\{x\}$ to y such that $\omega(t_w) < w$; in particular, this holds for $w = \alpha(y) - \alpha(x)$. As $\alpha \models S$, by Lemma 3 we have $\alpha(x) + \omega(t_w) \geqslant \alpha(y)$, i.e., $\omega(t_w) \geqslant \alpha(y) - \alpha(x)$, which is a contradiction.

Finally, as in $H(S)$ all vertices are hyperconnected to $\{x\}$ by a hyperedge, it is clear that δ_x is well-defined if and only if so is δ_X for all $X \subseteq V$, $X \neq \emptyset$.

Secondly, let us prove that given a hypergraph H, one can compute in polynomial time a set $S(H)$ of max-atoms such that H has a well-defined distance function δ_X for all $X \subseteq V$, $X \neq \emptyset$ if and only if $S(H)$ is satisfiable. Given $H = (V, E, W)$, the variables of $S(H)$ are V, the vertices of H; and for each hyperedge $(Z, y) \in E$, we consider the max-atom $\max_{z\in Z}(z) + W(Z, y) \geqslant y$. The proof concludes by observing that H has a well-defined distance function δ_X for all $X \subseteq V$, $X \neq \emptyset$ if and only if the same property holds for $H(S(H))$, if and only if $S(H)$ is satisfiable. □

5. From $\mathbb{Z}$ to $\mathbb{Q}$

If the domain of variables and offsets is $\mathbb{Q}$, one can naturally transform the original problem into an equivalent one in $\mathbb{Z}$ as follows. Given a conjunction of n atoms with rational offsets $\max(x_i, y_i) + p_i/q_i \geqslant z_i$, for i in $1 \ldots n$, if *lcm* is the least common multiple of the q_i's, one can express each atom as $\max(x_i, y_i) + r_i/lcm \geqslant z_i$ for certain r_i's and solve the equisatisfiable conjunction of atoms $\max(x_i, y_i) + r_i \geqslant z_i$ over $\mathbb{Z}$.

This shows that any PTIME algorithm for MAP over $\mathbb{Z}$ would also give us a PTIME algorithm over $\mathbb{Q}$. But this is not the case for the following *weakly* polynomial algorithm for MAP over $\mathbb{Z}$. W.l.o.g. in what follows max-atoms are of the form $\max(x, y) + k \geqslant z$ with $x \neq z$, $y \neq z$. This can be assumed by removing trivial contradictions $\max(x, x) + k \geqslant x$ ($k < 0$), trivial tautologies $\max(x, y) + k \geqslant x$ ($k \geqslant 0$), and by replacing $\max(x, y) + k \geqslant x$ by $\max(y, y) + k \geqslant x$ if $k < 0$ and $x \neq y$.

Definition 3. Given a set of max-atoms S defined over the variables V and two assignments α, α', we write $\alpha \to_S \alpha'$ (or simply $\alpha \to \alpha'$, if S is understood from the context) if there is a max-atom $\max(x, y) + k \geqslant z \in S$ such that:

1. $\alpha'(z) = \max(\alpha(x), \alpha(y)) + k$;
2. $\alpha'(z) < \alpha(z)$ (hence we say that z *decreases* in this step);
3. $\alpha'(u) = \alpha(u)$ for all $u \in V$, $u \neq z$.

Any sequence of steps $\alpha_0 \to \alpha_1 \to \cdots$ is called a *max-derivation* for S.

Lemma 4. *Let S be a set of max-atoms defined over the variables V. An assignment $\alpha : V \to \mathbb{Z}$ is a model for S if and only if α is* final, *i.e., there is no α' such that $\alpha \to \alpha'$.*

The following lemma expresses that max-derivations, while decreasing variables, never "break through" any model:

Lemma 5. *Let S be a set of max-atoms and let α be a model of S. If $\alpha_0 \to \cdots \to \alpha_m$ and $\alpha_0 \geqslant \alpha$, then $\alpha_m \geqslant \alpha$.*

Proof. By induction over m, the length of the derivation. For $m = 0$ there is nothing to prove. Now, if $m > 0$ the step $\alpha_0 \to \alpha_1$ is by an atom $\max(x, y) + k \geqslant z$. Let us prove that $\alpha_1 \geqslant \alpha$. We only need to show that the inequality holds for the variable that changes, which is z; and indeed $\alpha_1(z) = \max(\alpha_0(x), \alpha_0(y)) + k \geqslant \max(\alpha(x), \alpha(y)) + k \geqslant \alpha(z)$. Now, by induction hypothesis $\alpha_m \geqslant \alpha$. □

The next lemma ensures that models of a set of max-atoms are invariant under "uniform" translations:

Lemma 6. *Given a set of max-atoms S defined over the variables V and an assignment $\alpha : V \to \mathbb{Z}$ which is a model of S, for any $d \in \mathbb{Z}$ the assignment α' defined by $\alpha'(x) = \alpha(x) + d$ is a model of S.*

The previous lemmas, together with the Small Model Property, provide us with a weakly polynomial algorithm (i.e., runtime is polynomial if numbers are encoded in unary). This weakly polynomial algorithm can be seen as an extension of the Bellman–Ford algorithm for shortest paths (this also applies to the one of [9]):

Theorem 3. *MAP over $\mathbb{Z}$ is weakly polynomial.*

Proof. Let S be a conjunction of max-atoms, with variables V, where $|V| = n$. For deciding the satisfiability of S one can construct an arbitrary max-derivation, starting, e.g., from the assignment α_0 with $\alpha_0(x) = 0$ for all x in V. At each step, one variable decreases by at least one. If S is

Please cite this article in press as: M. Bezem et al., Hard problems in max-algebra, control theory, hypergraphs and other areas, Information Processing Letters (2009), doi:10.1016/j.ipl.2009.11.007

satisfiable, by the Small Model Property and by Lemma 6, there is a model α such that $-K_S \leqslant \alpha(x) \leqslant 0$ for all x in V. Moreover, by the previous lemma, no variable x will ever get lower than $\alpha(x)$ in the derivation. Altogether this means that, if no model is found after $n \cdot K_S$ steps, then S is unsatisfiable. □

As a corollary of the proof of the previous theorem, we obtain a PTIME decision procedure for sets of atoms of the forms $\max(x, y) \geqslant z$ or $\max(x, y) > z$. More generally, this also applies to *K-bounded sets*, where in S the absolute values of all offsets are bounded by a given constant K.

Example 2. Let S be the set of max-atoms $\{u - 10 \geqslant x, z \geqslant y, \max(x, y) - 1 \geqslant z, \max(x, u) + 25 \geqslant z\}$, and let α_0 be the assignment with $\alpha_0(x) = \alpha_0(y) = \alpha_0(z) = \alpha_0(u) = 0$. This initial assignment α_0 violates $u - 10 \geqslant x$, which allows us to decrease x and assign it the value -10: in terms of max-derivations $\alpha_0 \to \alpha_1$, where α_1 is the assignment with $\alpha_1(x) = -10$, $\alpha_1(y) = \alpha_1(z) = \alpha_1(u) = 0$.

Now the assignment α_1 only violates $\max(x, y) - 1 \geqslant z$, which forces z to take the value -1: in terms of max-derivations, $\alpha_1 \to \alpha_2$, where α_2 is the assignment with $\alpha_2(x) = -10$, $\alpha_2(y) = 0$, $\alpha_2(z) = -1$, $\alpha_2(u) = 0$. Then α_2 only violates $z \geqslant y$, which forces y to take the value -1 too: $\alpha_2 \to \alpha_3$, where α_3 is the assignment with $\alpha_3(x) = -10$, $\alpha_3(y) = \alpha_3(z) = -1$, $\alpha_3(u) = 0$.

It is easy to see that 11 iterations of each of the last two steps will be needed to find a model: finally we will have a derivation $\alpha_0 \to^* \alpha$ with $\alpha(x) = -10$, $\alpha(y) = \alpha(z) = -11$, $\alpha(u) = 0$; since there is no α' such that $\alpha \to \alpha'$, α is a model of S, hence S is satisfiable.

Notice that, if we replace 10 in S by larger powers of 10, we get a family of inputs whose sizes increase linearly, but for which the number of steps of the max-derivations reaching to a model grows exponentially. Since the number of steps is polynomial in the *value* of the offsets, and not in the *sizes* of the offsets, the algorithm based is weakly polynomial (but not polynomial).

Now, if we consider the set of max-atoms $S' = S \cup \{\max(x, y) + 9 \geqslant u\}$, we note that α above does not satisfy the new constraint. So we can decrease u and assign it the value -1, which makes $u - 10 \geqslant x$ false and forces x to take the value -11. Then $\max(x, y) - 1 \geqslant z$ is violated, and z is decreased to -12. Finally $z \geqslant y$ becomes false, so y is assigned -12. The loop of these four steps can be repeated over and over, making all variables decrease indefinitely. Thus, S' is unsatisfiable as no model is found within the bound of $n \cdot K_S$ steps given in the previous theorem.

The above transformation for MAP over $\mathbb{Z}$ into MAP over $\mathbb{Q}$ may produce an exponential blow-up in the *value* of the offsets. Thus, one cannot directly conclude that MAP over $\mathbb{Q}$ is weakly polynomial given that MAP over $\mathbb{Z}$ is so. Indeed, we have the following.

Theorem 4. *The algorithm given in the proof of Theorem* 3 *is not weakly polynomial for MAP over* $\mathbb{Q}$.

Proof. Let us fix $n \geqslant 1$ and $k \geqslant 0$. Then we define $c_j = \binom{k}{2j}\frac{1}{n+2j}$ for $0 \leqslant j \leqslant \lfloor k/2 \rfloor$; and $c_{\lfloor k/2 \rfloor + j} = \binom{k}{2j-1}\frac{-1}{n+2j-1}$ for $0 < j \leqslant \lceil k/2 \rceil$. Note that, by a simple induction on k, we have $\sum_{j=0}^{k} c_j = \sum_{i=0}^{k}\binom{k}{i}\frac{(-1)^i}{n+i} = \frac{k!(n-1)!}{(n+k)!}$.

Now let us assume that the algorithm is weakly polynomial over $\mathbb{Q}$ and we will get a contradiction. Let d be such that, given S a satisfiable instance of MAP over $\mathbb{Q}$, *any* max-derivation for S has length at most $O(\text{size}(S)^d)$, where numbers are represented in unary. Let us fix k such that $k > d$. Consider the following family of instances, parameterized by n:

$$\begin{aligned} S_n = \big\{ \max(y - 1 - c_0, x_0 - c_0) &\geqslant x_1 \\ x_1 - c_1 &\geqslant x_2 \\ &\cdots \\ x_i - c_i &\geqslant x_{i+1} \\ &\cdots \\ x_{k-1} - c_{k-1} &\geqslant x_k \\ x_k - c_k &\geqslant x_0 \big\}. \end{aligned}$$

Each S_n is satisfiable, as e.g. the assignment $\alpha(y) = 1 + \sum_{j=0}^{k} c_j$, $\alpha(x_0) = 0$, $\alpha(x_i) = \sum_{j=i}^{k} c_j$ for $1 \leqslant i \leqslant k$ is a model.

Let us apply the algorithm starting from the assignment that maps all variables to 0. As $\max(y - 1 - c_0, x_0 - c_0) \geqslant x_1$ is not satisfied, one can apply a derivation step and update the value of x_1 to $-c_0$. Next, as $c_0 + c_1 > 0$, the atom $x_1 - c_1 \geqslant x_2$ is not satisfied, and one can update the value of x_2 to $-(c_0 + c_1)$. In general, as positive c_j come first, $\sum_{j=0}^{i} c_j > 0$ for $0 \leqslant i \leqslant k$; hence all variables $x_3, \ldots, x_k$, x_0 can be updated, in this order. The new value of x_0 is $-\sum_{j=0}^{k} c_j = -\frac{k!(n-1)!}{(n+k)!}$.

Now the loop of updating variables $x_1, \ldots, x_k, x_0$ can be repeated until x_0 gets the value -1 (after one more round the model is found). As x_0 is decreased by $-\frac{k!(n-1)!}{(n+k)!}$ after each loop, this will happen after $\frac{(n+k)!}{k!(n-1)!}$ loops. But as n increases, this number grows as a polynomial of degree $k > d$, while $\text{size}(S_n)$ grows linearly. This yields a contradiction.

Note that, in fact, the DL-atoms with positive offsets can only be used in the order in which they are enumerated in S_n. Since the absolute value of every c_j is larger than $\frac{k!(n-1)!}{(n+k)!}$ (for big enough n), one has to use all DL-atoms with negative offsets to achieve a decrease of x_0. Hence *any* other max-derivation takes at least as many steps as the one described above. □

Acknowledgements

We would like to thank Daniele Pretolani for insightful comments on hypergraphs, an anonymous referee of [3] for suggesting the equivalence with super fixpoints of min–max functions and possible connections with game theory, which got confirmed when Albert Atserias and Elitza Maneva (whom we also thank!) made us aware of [9].

References

[1] Giorgio Ausiello, Giuseppe F. Italiano, Umberto Nanni, Optimal traversal of directed hypergraphs, Technical Report TR-92-073 ICSI, Berkeley, CA, 1992.

[2] Marc Bezem, Robert Nieuwenhuis, Enric Rodriguez-Carbonell, Exponential behaviour of the Butkovič–Zimmermann algorithm for solving two-sided linear systems in max-algebra, Discrete Applied Mathematics 156 (18) (2008) 3506–3509.

[3] Marc Bezem, Robert Nieuwenhuis, Enric Rodríguez-Carbonell, The max-atom problem and its relevance, in: Iliano Cervesato, Helmut Veith, Andrei Voronkov (Eds.), LPAR, in: Lecture Notes in Computer Science, vol. 5330, Springer, 2008, pp. 47–61.

[4] Peter Butkovič, Karel Zimmermann, A strongly polynomial algorithm for solving two-sided linear systems in max-algebra, Discrete Applied Mathematics 154 (3) (2006) 437–446.

[5] A. Ehrenfeucht, J. Mycielski, Positional strategies for mean payoff games, International Journal of Game Theory 8 (2) (1979) 109–113.

[6] E. Allen Emerson, Charanjit S. Jutla, A. Prasad Sistla, On model-checking for fragments of μ-calculus, in: Costas Courcoubetis (Ed.), CAV, in: Lecture Notes in Computer Science, vol. 697, Springer, 1993, pp. 385–396.

[7] Giorgio Gallo, Giustino Longo, Stefano Pallottino, Sang Nguyen, Directed hypergraphs and applications, Discrete Applied Mathematics 42 (1993) 177–201.

[8] J. Gunawardena, Min–max functions, Discrete Events Dynamic Systems 4 (1994) 377–406.

[9] Rolf H. Möhring, Martin Skutella, Frederik Stork, Scheduling with AND/OR precedence constraints, SIAM Journal on Computing 33 (2) (2004) 393–415.

[10] Robert Nieuwenhuis, Albert Oliveras, Cesare Tinelli, Solving SAT and SAT modulo theories: From an abstract Davis–Putnam–Logemann–Loveland procedure to DPLL(T), Journal of the ACM 53 (6) (November 2006) 937–977.

Please cite this article in press as: M. Bezem et al., Hard problems in max-algebra, control theory, hypergraphs and other areas, Information Processing Letters (2009), doi:10.1016/j.ipl.2009.11.007