21st International Symposium on Mathematical Theory of Networks and Systems
July 7-11, 2014. Groningen, The Netherlands

New Transience Bounds for Long Walks

Bernadette Charron-Bost[1] Matthias Függer[2] Thomas Nowak[3]

***Abstract*— Linear max-plus systems describe the behavior of a large variety of complex systems. It is known that these systems show a periodic behavior after an initial transient phase. Assessment of the length of this transient phase provides important information on the performance of such systems, and so is crucial in system design. We identify relevant parameters in a graph representation of these systems and propose a modular strategy to derive new upper bounds on the length of the transient phase. By that we are the first to give asymptotically tight and potentially subquadratic transience bounds. We use our bounds to derive new complexity results, in particular in distributed computing.**

I. Introduction

The behavior of many complex systems can be described by a sequence of N-dimensional vectors $x(n)$ that satisfy a recurrence relation of the form

$$\forall n \geqslant 1 \ \forall i \in [N] \colon x_i(n) = \max_{j \in \mathbf{N}_i} \left(x_j(n-1) + A_{i,j} \right) \quad (1)$$

where the $A_{i,j}$ are real numbers, and the $\mathbf{N}_i$ are subsets of $\{1, \ldots, N\}$. For instance, $x_i(n)$ may represent the time of the nth occurence of a certain event i and $A_{i,j}$ the required time lag between the $(n-1)$th occurence of j and the nth occurence of i. Notable examples are transportation and automated manufacturing systems [16], [10], network synchronizers [17], [12], and cyclic scheduling [14]. Recently, Charron-Bost et al. [6], [7] have shown that it also encompasses the behavior of an important class of distributed algorithms, namely *link reversal algorithms* [13], which can be used to solve a variety of problems [19] like routing [13] or resource allocation [5].

Recurrences of the form (1) are linear in the max-plus algebra. The fundamental theorem in max-plus linear algebra—an analog of the Perron-Frobenius theorem—states that the sequence of powers of an irreducible matrix A becomes periodic after a finite index called the *transient*, or sometimes the *coupling time* [16], of the matrix. As an immediate corollary, any max-plus linear system with an irreducible matrix is periodic from some index, called the *transient* of the system, which clearly is at most equal to the transient of the system's matrix. For all the above mentioned applications, the study of the transient plays a key role in characterizing the system performances: For example, in the case of link reversal routing, the system transient corresponds to the time complexity of the routing algorithm. Besides that, understanding matrix and system transients is of interest on its own for the theory of max-plus algebra.

Hartmann and Arguelles [15] have shown that the transients of matrices and linear systems are computable in polynomial time. However, their algorithms provide no analysis of the transient phase, and do not hint at the parameters that influence matrix and system transients. Upper bounds involving these parameters help to predict the duration of the transient phase, and to define strategies to reduce transients during system design. From both numerical and methodological viewpoints, it is therefore important to determine accurate transience bounds.

In this paper, we present two new upper bounds on the transients of linear max-plus systems. Our approach is graph-theoretic in nature: The problem of upper bounding the transient can be reduced to the study of walks in a specific graph. More precisely, for every matrix A, one considers the weighted directed graph G whose adjacency matrix is A, and its *critical subgraph* G_c consisting of the *critical cycles*, namely those cycles with maximum mean weight. The entries of the max-plus matrix power $A^{\otimes n}$ are equal to the maximum weights of walks in G of length n between two fixed nodes. Periodicity stems from the fact that eventually the weights of critical cycles dominate the maximum weight walks.

We present a modular graph-based strategy whose core idea is a walk reduction $\mathrm{Red}_{d,k}$, which removes cycles from a walk while assuring that the resulting length remains in the same residue class modulo d, and that node k rests on the walk. The key property of $\mathrm{Red}_{d,k}$ is an upper bound on the length of the reduced walk that is linear both in d and the number of nodes in the graph. The following step in our strategy consists in completing reduced walks with critical cycles of appropriate lengths. To show existence of critical cycles of predefined length, we propose two methods, namely the *repetitive* method and the *explorative* method. In the first one, the visit of the critical subgraph is confined to repeatedly follow only one critical cycle, which leads us to choosing d equal to the girth (shortest cycle length) of a specific strongly connected component of the critical subgraph. The second one consists in exploring one whole strongly connected component of the critical subgraph, and leads us to choosing d equal to the component's cyclicity (greatest common divisor of cycle lengths). Then, we use the notion of the *exploration penalty* of a strongly connected graph G as the least integer k with the property that, for every $n \geqslant k$ divisible by the cyclicity of G and every node i of G, there is a closed path starting and ending at i of length n. One key point is an upper bound on the exploration penalty—

[1]Bernadette Charron-Bost is with the CNRS, École polytechnique, Palaiseau, France. Email: charron@lix.polytechnique.fr
[2]Matthias Függer is with the Vienna University of Technology, Vienna, Austria. Email: fuegger@ecs.tuwien.ac.at
[3]Thomas Nowak is with the École normale supérieure, Paris, France. When this work was initiated, he was with the École polytechnique, Palaiseau, France. Email: thomas.nowak@ens.fr

ISBN: 978-90-367-6321-9

derived in the full paper [8]—which is at most quadratic in the number of nodes.

We hence prove the following two upper bounds on the transient of N-dimensional max-plus linear systems with irreducible system matrix A and initial vector v with finite entries. The terms $\|A\|$ and $\|v\|$ denote the difference of the maximum and minimum finite entries of A and v, respectively. The mean weight of critical cycles is denoted by λ, and λ_{nc} denotes the largest mean weight of cycles that have no node on critical cycles.

Theorem 1 (Repetitive Bound): Denoting by $\hat{g}$ the maximum girth of strongly connected components of G_c, the transient of the max-plus linear system defined by A and v is at most

$$\max\left\{\tfrac{\|v\|+\|A\|\cdot(N-1)}{\lambda-\lambda_{\mathrm{nc}}}\ ,\ (\hat{g}-1)+2\,\hat{g}\cdot(N-1)\right\}\ .$$

Theorem 2 (Explorative Bound): Denoting by $\hat{\gamma}$ and $\hat{\mathrm{ind}}$ the maximum cyclicity and maximum exploration penalty of strongly connected components of G_c, respectively, the transient of the max-plus linear system defined by A and v is at most

$$\max\left\{\tfrac{\|v\|+\|A\|\cdot(N-1)}{\lambda-\lambda_{\mathrm{nc}}}\ ,\ (\hat{\gamma}-1)+2\,\hat{\gamma}\cdot(N-1)+\hat{\mathrm{ind}}\right\}\ .$$

The repetitive and explorative bounds are incomparable in general. We show that in the case of integer matrices, for a given initial vector, both bounds are in $O(\|A\|\cdot N^3)$, and that our transience bounds are asymptotically tight.

The problem of bounding transients has already been studied (e.g., [15], [3], [18]), and the best previously known bounds have been given by Hartmann and Arguelles [15]. Their bound on system transients is, in general, incomparable with both our repetitive and explorative bounds. The significant benefit of our two new bounds is that each of them turns out to be linear in the size of the system in various classes of linear max-plus systems whereas Hartmann and Arguelles' bound is intrinsically at least quadratic. This is mainly due to the introduction of graph parameters like the girth, cyclicity, and exploration penalty that enables a fine-grained analysis of the transient phase. In particular, that helps system designers in the choice of specific topologies, reducing system transients.

Finally, we demonstrate how our transience bounds enable the performance analysis of a large variety of distributed systems. First, we derive two transience bounds for a large class of synchronizers, and we quantify how our synchronizer bounds are better than that given by Even and Rajsbaum [12] in their specific case of integer delays. From this synchronizer example, we show that our general transience bounds are both asymptotically tight. Then we apply our bounds to the analysis of the performance of distributed routers and schedulers based on the link-reversal algorithms: We obtain $O(N^3)$ transience bounds, improving the $O(N^4)$ bound established by Malka and Rajsbaum [17], and new $O(N)$ bounds for such routers and schedulers when running in trees.

II. Proof Strategy Outline

We start by defining for a set $\mathbf{N}$ of nonnegative integers and a node i, an $\mathbf{N}$-*realizer for node* i to be a walk of maximum A_v-weight in the set of walks in $\mathcal{W}(i\to)$ with length in $\mathbf{N}$. As shown in Proposition 3, of particular interest is the case of sets $\mathbf{N}$ of the form

$$\mathbf{N}_{\geq B}^{(n,\pi)}=\{m\in\mathbb{N}\mid m\geqslant B\ \wedge\ m\equiv n\pmod{\pi}\}$$

where B, n, and π are positive integers.

Proposition 3: Let B and π be positive integers. If there exists, for each node i and each integer $n\geqslant B$, an $\mathbf{N}_{\geq B}^{(n,\pi)}$-realizer for i of length n, then B is an upper bound on the system transient.

Based on Proposition 3, we now define a strategy for determining upper bounds on system transients. Let $n\in\mathbb{N}$, and let i be any node. Denote by π the least common multiple of cycle lengths in the critical subgraph G_{c}. The strategy includes an additional parameter B to be chosen in step 4 so that we can construct an $\mathbf{N}_{\geq B}^{(n,\pi)}$-realizer if $n\geq B$.

1) *Normalized matrix.* Because the transients of A and of $\overline{A}$ are equal, and the ratio of $\overline{A}$ is 0, we can reduce the general case to the case $\lambda=0$. The latter condition guarantees the existence of realizers for every nonempty $\mathbf{N}\subseteq\mathbb{N}$ and yields that adding critical cycles to a walk does not change its A-weight. The rest of the strategy hence considers an irreducible matrix A with $\lambda=0$. Let W be an $\mathbf{N}_{\geq B}^{(n,\pi)}$-realizer for node i.
2) *Critical bound.* We show that for B large enough, i.e., B greater or equal to some *critical bound* B_{c}, the realizer W contains at least one critical node k.
3) *Walk reduction.* Next we show that for every divisor d of π, by removing subcycles, we can *reduce* W to a new walk $\hat{W}$ which starts at node i, contains the critical node k, whose length $\ell(\hat{W})$ is in the same residue class modulo d as $\ell(W)$, and $\ell(\hat{W})$ is upper-bounded by a term linear in the number of nodes in the graph.
4) *Pumping in the critical graph.* Since d divides π, d divides $n-\ell(\hat{W})$, and for two appropriate choices of d and for n sufficiently large ($n\geq B_d$), we show how to complete $\hat{W}$ by adding to it a critical closed walk starting from k in order to obtain a new walk of length n starting at node i. For $B=\max\{B_{\mathrm{c}},B_d\}$, this yields an $\mathbf{N}_{\geq B}^{(n,\pi)}$-realizer of length n, because removing cycles at most increases the weight and adding a critical closed path does not change the weight. Proposition 3 then shows that B is a bound on the transient.

III. Applications

In this section we demonstrate how our transience bounds enable the performance analysis of various distributed systems, thereby obtaining simple proofs both of known and new results. One of the given examples shows the asymptotic tightness of our bounds.

A. Synchronizers

Even and Rajsbaum [12] presented a transience bound for a network synchronizer in a system with constant integer communication delays. They considered a variant of the α-synchronizer [1] in a centrally clocked distributed system of N processes that communicate by message passing over a strongly connected network graph G. Each link has constant transmission delay, specified in terms of central clock ticks. Processes execute the α-synchronizer after an initial boot-up phase: After receiving round n messages from all neighbors, a process proceeds to round $n+1$ and broadcasts its round $n+1$ message. Denote by $t(n)$ the vector such that $t_i(n)$ is the clock tick at which process i broadcasts its round n message. Even and Rajsbaum showed that the synchronizer becomes periodic by time $B_{\text{ER}} = l_0 + 2N^2 + N$, where l_0 is an upper bound on the length of maximum weight walks with only non-critical nodes. It is easily checked that l_0 is always greater or equal to our critical bound B_{c}.

One can show that $t(n)$ is in fact a max-plus linear system. More precisely, $t(n) = A^{\otimes n} \otimes t(0)$, where A is the adjacency matrix of the network graph G. Our bounds hence directly apply, and we obtain a repetitive bound on the transient of $(t(n))_{n\geq 0}$ that is strictly less than $\max\{l_0, 2N^2 - N\}$, and thus strictly less than Even and Rajsbaum's bound B_{ER}.

We next show asymptotic tightness of our transience bounds. Consider the "ℓ-sized cherry" graph family $H_{\ell,c}$, with $\ell \geq 2$ and $c \geq 1$, introduced by Even and Rajsbaum [12]. Each weighted graph $H_{\ell,c}$ contains $N = 4\ell$ nodes and is constructed as follows: Let $\hat{C}$ and C be two cycles of length ℓ and $\ell+1$ respectively, with edge weights $3c$, except for one edge per cycle with weight $3c+1$. There exists for both $\hat{C}$ and C a path of length ℓ to a distinct node s, and an antiparallel path back. Hereby the edges in the path from s to C and from s to $\hat{C}$ have weight c, the edges in the path from $\hat{C}$ to s have weight $3c$, and from C to s, $4c$. Since Even and Rajsbaum expressed transmission delays with respect to a discrete global clock, all weights are integers. Thus both our transience bounds are in $O(\|A\| \cdot N^3)$, showing their asymptotic tightness since Even and Rajsbaum proved that the transient for graph $H_{c,\ell}$ is in $\Omega(c \cdot \ell^3) = \Omega(\|A\| \cdot N^3)$.

B. Full Reversal routing and scheduling

Link reversal is a versatile algorithm design paradigm, which was, in particular, successfully applied to routing [13] and scheduling [2]. Charron-Bost et al. [7] showed that the analysis of a general class of link reversal algorithms can be reduced to the analysis of Full Reversal, a particularly simple algorithm on directed graphs.

The Full Reversal algorithm comprises only a single rule: Each sink reverses all its (incoming) edges. Given a weakly connected initial graph G_0 without antiparallel edges, we consider a *greedy* execution of Full Reversal as a sequence $(G_t)_{t\geq 0}$ of graphs, where G_{t+1} is obtained from G_t by reversing the edges of *all* sinks in G_t. As no two sinks in G_t can be adjacent, G_{t+1} is well-defined. For each $t \geq 0$ we define the *work vector* $W(t)$ by setting $W_i(t)$ to the number of reversals of node i until iteration t, i.e., the number of times node i is a sink in the execution prefix $G_0, \ldots, G_{t-1}$.

Charron-Bost et al. [6] have shown that the sequence of work vectors can be described as a *min-plus* linear system. Denoting by $\otimes'$ the matrix multiplication in min-plus algebra, Charron-Bost et al. established that $W(0) = 0$ and $W(t+1) = A \otimes' W(t)$, where $A_{i,j} = 1$ and $A_{j,i} = 0$ if (i,j) is an edge of the initial graph G_0; otherwise $A_{i,j} = +\infty$. Observe that the latter min-plus recurrence is equivalent to $-W(t+1) = (-A) \otimes (-W(t))$, where $-A$ is an integer max-plus matrix with $\|A\| \leq 1$.

Full Reversal routing. In the routing case, the initial graph G_0 contains a nonempty set of *destination nodes*, which are characterized by having a self-loop. The initial graph without these self-loops is required to be weakly connected and acyclic [6], [13]. It was shown that for such initial graphs, the execution terminates (eventually all G_t are equal), and after termination, the graph is destination-oriented, i.e., every node has a walk to some destination node. We now show how the previously known results that the termination time of Full Reversal routing is quadratic in general [4] and linear in trees [6] directly follows from Theorem 1 or Theorem 2.

The set of critical nodes is equal to the set of destination nodes and each strongly connected component of G_{c} consists of a single node. Hence $\lambda = 0$ and $\lambda_{\text{nc}} \leq -1/N_{\text{nc}} \leq -1/(N-1)$, i.e., $(N-1)^2$ is an upper bound on the critical bound. Since $\hat{g} = \hat{\gamma} = 1$, we obtain from Theorem 1 or Theorem 2, for $N \geqslant 3$, that the termination time is at most $(N-1)^2$, which improves on the asymptotic quadratic bound given by Busch and Tirthapura [4].

If the undirected support of initial graph G_0 without the self-loop at the destination nodes is a *tree*, we can use our bounds to give a new proof that the termination time of Full Reversal routing is linear in N [6]. In that particular case either $\lambda_{\text{nc}} = -1/2$ or $\lambda_{\text{nc}} = -\infty$. In both cases the critical bound is at most $2(N-1)$. Both Theorem 1 and Theorem 2 yield the linear bound $2(N-1)$, whereas Hartmann and Arguelles arrive at $2N^2$.

Full Reversal scheduling. When using the Full Reversal algorithm for scheduling, the undirected support of the weakly connected initial graph G_0 is interpreted as a conflict graph: nodes model processes and an edge between two processes signifies the existence of a shared resource whose access is mutually exclusive. The direction of an edge signifies which process is allowed to use the resource next. A process waits until it is allowed to use all its resources—i.e., it waits until it is a sink—and then performs a step,i.e., reverses all edges to release its resources. To guarantee liveness, the initial graph G_0 is required to be acyclic.

Contrary to the routing case, strongly connected components of the critical subgraph have at least two nodes, because there are no self-loops. We get $N^2(N-1)/4$ as an upper bound on our critical bound, which shows that the transient for Full Reversal scheduling is in $O(N^3)$. Malka and Rajsbaum [17] proved by reduction to Timed Marked Graphs that the transient is at most in the order of $O(N^4)$. Thus, our bounds allow to improve their asymptotic result by an order of N.

In the case of *trees* we again have $\lambda = -1/2$ and $\lambda_{\mathrm{nc}} = -\infty$. Thus the critical bound is N. Further, $G_{\mathrm{c}} = G$ and $\hat{g} = \hat{\gamma} = 2$. Both Theorem 1 and Theorem 2 thus imply that $4N - 3$ is an upper bound on the transient of Full Reversal scheduling on trees, which is linear in N. This was previously unknown. By contrast Hartmann and Arguelles again obtain the quadratic bound of $2N^2$.

C. Cyclic scheduling

Cohen et al. [11] have observed that, in cyclic scheduling, the class of *earliest schedules* can be described as max-plus linear systems. In this section, we show how to use this fact and our general bounds to derive explicit upper bounds on transients of earliest schedules.

If a finite set $\mathcal{T}$ of tasks (each of which calculates a certain function) is to be scheduled repeatedly on different processes, precedence restrictions are implied by the data flow. These restrictions are of the form that task i may start its number n execution only after task j has finished its number $n - h$ execution. A *schedule* t maps a pair $(i, n) \in \mathcal{T} \times \mathbb{N}$ to a nonnegative integer $t(i, n)$, the time the number n execution of task i is started. Formally, if P_i denotes the processing time of task i, then a *restriction* R between two tasks i and j is an inequality of the form

$$\forall n \geqslant h_R : \; t(i, n) \geq t(j, n - h_R) + P_j \qquad (2)$$

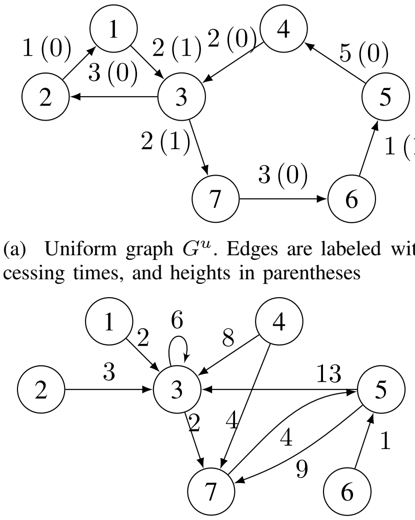
where h_R is called the *height* R and P_j is the *weight*.

A *uniform graph* [14] describes a set of tasks and restrictions. Formally, it is a quadruple $G^u = (\mathcal{T}, E, p, h)$ such that $(\mathcal{T}, E)$ is a directed (multi-)graph, and $p : E \to \mathbb{N}^*$ and $h : E \to \mathbb{N}$ are two functions, the *weight* and *height* function, respectively. For a walk W in G^u, let $p(W)$ be the sum of the weights of its edges and $h(W)$ the sum of the heights of its edges. An edge from i to j corresponds to a restriction R between i and j of the form (2). All incoming edges of a node j in $\mathcal{T}$ have the same weight, namely P_j.

We may now directly apply Theorems 1 and 2 to the strongly connected digraph G, obtaining upper bounds on the transients of the earliest schedule for G^u. For the given example, 106 is an upper bound on the transient of the earliest schedule. Relating parameters of the graphs G^u and G, we have

$$\lambda(G) = \max\{p(C)/h(C) \mid C \text{ is a closed walk in } G^u\} \ .$$

Our bounds then show that the transient is $O(|\mathcal{T}|^4)$, assuming constantly bounded weights in G^u. This is the first asymptotic bound on the transient of an earliest schedule with tasks $\mathcal{T}$ and binary heights. We pose the asymptotic tightness of this bound as an open problem.



(a) Uniform graph G^u. Edges are labeled with processing times, and heights in parentheses

(b) Graph G

Fig. 1. Example of a set of tasks with restrictions

References

[1] B. Awerbuch. Complexity of network synchronization. *J. ACM* **32**(4) 804–823, 1985.

[2] V.C. Barbosa, E. Gafni. Concurrency in heavily loaded neighborhood-constrained systems. *ACM Trans. Program. Lang. Syst.* **11**(4) 562–584, 1989.

[3] A. Bouillard, B. Gaujal. Coupling time of a (max,plus) matrix. *Proc. 1st IFAC Symposium on System Structure and Control*, Elsevier, Amsterdam, 2001.

[4] C. Busch, S. Tirthapura. Analysis of link reversal routing algorithms. *SIAM J. Comput.* **35**(2) 305–326, 2005.

[5] K.M. Chandy, J. Misra. The drinking philosophers problem. *ACM Trans. Program. Lang. Syst.* **6**(4) 632–646, 1984.

[6] B. Charron-Bost, M. Függer, J.L. Welch, J. Widder. Full reversal routing as a linear dynamical system. *Proc. 18th Int. Coll. Structural Information and Communication Complexity* (*SIROCCO* 2011). LNCS 6796, Springer, Heidelberg, 2011, pp. 101–112.

[7] B. Charron-Bost, M. Függer, J.L. Welch, J. Widder. Partial is full. *Proc. 18th Int. Coll. Structural Information and Communication Complexity* (*SIROCCO* 2011). LNCS 6796, Springer, Heidelberg, 2011, pp. 113–124.

[8] B. Charron-Bost, M. Függer, T. Nowak. New transience bounds for long walks, 2012. `arXiv:1209.3342v1 [cs.DM]`

[9] G. Cohen, D. Dubois, J.-P. Quadrat, M. Viot. Analyse du comportement périodique de systèmes de production par la théorie des dioïdes. INRIA Research Report 191, 1983.

[10] G. Cohen, D. Dubois, J.-P. Quadrat, M. Viot. A linear system-theoretic view of discrete event processes and its use for performance evaluation in manufacturing. *IEEE Trans. Automat. Control* **30**(3) 210–220, 1985.

[11] G. Cohen, P. Moller, J.-P. Quadrat, M. Viot. 1989. Algebraic tools for the performance evaluation of discrete event systems. *P. IEEE* **77**(1) 39–85, 1989.

[12] S. Even, S. Rajsbaum. The use of a synchronizer yields maximum computation rate in distributed systems. *Theory Comput. Syst.* **30**(5) 447–474, 1997.

[13] E.M. Gafni, D.P. Bertsekas. Asymptotic optimality of shortest path routing algorithms. *IEEE Trans. Inform. Theory* **33**(1) 83–90, 1987.

[14] C. Hanen, A. Munier. Cyclic scheduling on parallel processes: an overview. *Scheduling Theory and its Applications*. John Wiley & Sons, Chichester, 1995.

[15] M. Hartmann, C. Arguelles. Transience bounds for long walks. *Math. Oper. Res.* **24**(2) 414–439, 1999.

[16] B. Heidergott, G.J. Olsder, J. van der Woude. *Max Plus at Work*. Princeton University Press, Princeton, 2006.

[17] Y. Malka, S. Rajsbaum. Analysis of distributed algorithms based on recurrence relations. *Proc. 5th Int. Workshop on Distributed Algorithms* (*WDAG* 1991). LNCS 579, Springer, Heidelberg, 1992, pp. 242–253

[18] G. Soto y Koelemeijer. *On the Behaviour of Classes of Min-Max-Plus Systems*. PhD Thesis, TU Delft, 2003.

[19] J.L. Welch, J.E. Walter. *Link Reversal Algorithms*. Morgan & Claypool, San Rafael, 2012.