COMPUTER SCIENCES DEPARTMENT

University of Wisconsin-Madison

PROPERTIES OF VECTOR ADDITION SYSTEMS*

by

Lawrence H. Landweber

Technical Report #258

June 1975

Properties of Vector Addition Systems*

by

Lawrence H. Landweber
Computer Sciences Department
University of Wisconsin
1210 W. Dayton Street
Madison, Wisconsin 53706

ABSTRACT

We consider vector addition systems (VAS), a model of asynchronous computation which is equivalent to the Petri net model. Reachability is not known to be decidable for arbitrary VAS. The best known lower bound for the general case is exponential space-hard [14]. We first show that reachability for conflict free nets is NP-hard. Our next result, using a reduction similar to that of Jones [5] for reachability, is a PSPACE-hard lower bound on the complexity of deciding various properties of VAS such as safeness, boundedness and persistence. Since safeness can be decided in polynomial space, this result characterizes the space complexity of this problem. The construction also yields a PSPACE-hard lower bound for persistent nets. The next result is a reduction of persistence to reachability, partially answering a question of Keller [8]. Reachability and boundedness are then proved to be undecidable for the time VAS, introduced by Merlin [9].

*This research was partially supported by NSF Contract GJ 33087.

1. Introduction

The properties of models of asynchronous computations have been widely studied in recent years [2,3,4,7,8,9,11,12]. We continue this study with particular emphases on vector addition systems (VAS). Since this model is equivalent to Petri nets [11] and vector replacement systems [8], our results, suitably translated, are also valid for these models. In section 3, we give a PSPACE-hard lower bound on the complexity of deciding various properties of VAS. We also consider reachability, showing that this problem is NP-hard for conflict free VAS and PSPACE-hard for persistent VAS. These lower bounds are the best known for these special types of VAS. For arbitrary VAS, Lipton [14] has shown that reachability is exponential space-hard. Conflict free VAS have a decidable reachability problem. It is not known whether reachability is decidable for arbitrary VAS. In section 4, we show that Keller's [8] persistence property is reducible to reachability. In [8], the decidability of persistence and its recursive relationship to reachability were left open. The last section considers a variant of VAS, introduced by Merlin [9], in which a weak timing mechanism is permitted and shows that reachability is undecidable for this model.

2. Definitions

Let N stand for the natural numbers, I for the integers. Let $v,\bar{v}$ be n-dimensional integer valued vectors. Then $(v)_i$ is the i-th coordinate of v. $v \geq \bar{v}$ if $(v)_i \geq (\bar{v})_i$ for $i = 1,\ldots,n$. $v \geq k$, if $(v)_i \geq k$ for $i = 1,\ldots,n$.

A Vector Addition System (VAS) $\mathcal{V}$ is a pair $\langle v_0,V\rangle$ where $v_0 \in N^n$ and $V = \{v_1,\ldots,v_r\}$ is a finite set of vectors in I^n. A vector $v \in N^n$ is reachable via $\mathcal{V}$ if there exist indices $i_1,\ldots,i_k$ $k \geq 0$, $1 \leq i_k \leq r$ such that

$$v_0 + v_{i_1} + \cdots + v_{i_k} = v$$

and moreover for all $j \leq k$

$$v_0 + v_{i_1} + \cdots + v_{i_j} \geq 0.$$

The reachability set $R(\mathcal{V})$ is the set of all vectors reachable via $\mathcal{V}$.

Note that the second condition in the definition of reachable requires that all intermediate points between v_0 and v are to be in the first orthant.

The reachability problem is the problem of deciding for arbitrary VAS $\mathcal{V}$ and vector v whether $v \in R(\mathcal{V})$. We may similarly define the reachability problem for special types of VAS.

Let $\mathcal{V} = \langle v_0,V\rangle$ and $v \in V$. Then $\bar{v} \xrightarrow[v]{}$ means that $\bar{v} + v \geq 0$, i.e., that v can be applied to $\bar{v}$. Similarly, for $v_1,\ldots,v_k \in V$, $\bar{v} \xrightarrow[v_1\ldots v_k]{}$ means $\bar{v} + v_1 + \cdots + v_j \geq 0$ for $1 \leq j \leq k$.

We restrict our attention to VAS $V = \langle v_0, V\rangle$ in which all coordinates of each $v \in V$ are members of $\{-1,0,+1\}$. This is an inessential change since all VAS can be modified, via the use of additional coordinates, to satisfy this condition. (Though of course this changes the set of reachable points.)

A VAS V is <u>bounded</u> if there is a $\bar{v} \in N^n$ such that $v \le \bar{v}$ for every $v \in R(V)$. V is <u>safe</u> if it is bounded by 1.

A VAS V is <u>persistent</u> if $v \xrightarrow[v_i]{}$ and $v \xrightarrow[v_j]{}$, $v_i \neq v_j$ imply $v \xrightarrow[v_i v_j]{}$ for every $v \in R(V)$. V is <u>conflict free</u> if $v, \bar{v} \in V$ and $(v)_i = (\bar{v})_i = -1$ imply $v = \bar{v}$. Note that all conflict free VAS are persistent.

In the following, we assume familiarity with the ideas of [1,6,12]. In particular, the notions NP-hard, PSPACE-hard and polynomial time reducible are used. The reader is referred to the above references for definitions of these terms. Informally NP is the class of problems solvable on a non-deterministic Turing machine in polynomial time. PSPACE is the class of problems solvable with polynomial space on a Turing machine. A problem is NP-hard (PSPACE-hard, exponential space-hard) if every problem soluable in non-deterministic polynomial time (polynomial space, exponential soace) is polynomial time reducible to it.

3. <u>Complexity of VAS Problems</u>

It is not known whether $R(V)$ is recursive for arbitrary VAS. For conflict free [2], and arbitrary three coordinate [12] VAS, reachability is decidable. Our first theorem gives an NP-hard lower bound on reachability for conflict free, bounded and safe VAS. For safe (bounded) nets, this can be improved to PSPACE-hard [5]. Since reachability for safe VAS is decidable in polynomial space, the space complexity of reachability is known for safe VAS. For conflict free VAS, the lower bound of Theorem 1 is the best known.

<u>Theorem 1</u>. The reachability problem for conflict free, bounded and safe VAS is NP-hard.

<u>Proof</u>. The proof involves exhibiting a polynomial time reduction of the satisfiability problem for propositional calculus conjunctive normal form formulas [1,6] to the reachability problem. Given a CNF F, we build a conflict free VAS such that some point v is reachable if and only if F is satisfiable. Rather than include a detailed proof, we illustrate the construction via an example.

A. <u>Conflict free VAS</u>

Given the CNF formula

$$F = (\neg P \vee Q) \wedge (\neg P \vee \neg Q \vee R) \wedge Q \wedge (\neg Q \vee R \vee P)$$

build a conflict free VAS V_F as follows:

$$v_0 = 0$$

$$V = \{v_P, v_{\neg P}, v_Q, v_{\neg Q}, v_R, v_{\neg R}, v_1, v_2, v_3, v_4\}$$

where each member of V has seven coordinates, one for each conjunct of F and one for each of the atoms (P,Q,R) of F. Now

$$v_P = (0,0,0,1,1,0,0)$$

$$v_{\neg P} = (1,1,0,0,1,0,0)$$

$$v_Q = (1,0,1,0,0,1,0)$$

$v_{\neg Q} = (0,1,0,1,0,1,0)$

$v_R = (0,1,0,1,0,0,1)$

$v_{\neg R} = (0,0,0,0,0,0,1)$

$v_1 = (-1,0,0,0,0,0,0)$

$v_2 = (0,-1,0,0,0,0,0)$

$v_3 = (0,0,-1,0,0,0,0)$

$v_4 = (0,0,0,-1,0,0,0)$.

We show that $v = 1$ is in $R(V_F)$ iff F is satisfiable.

First note that v_P and $v_{\neg P}$ have 1 in the fifth coordinate; v_Q and $v_{\neg Q}$ have 1 in the sixth coordinate; v_R and $v_{\neg R}$ have 1 in the seventh coordinate. No vector of V_F has -1 in any of these coordinates. Hence to reach $v = 1$ from $v_0 = 0$ one and only one of $v_P, v_{\neg P}$ $(v_Q, v_{\neg Q})(v_R, v_{\neg R})$ must be added. Each choice of these vectors corresponds to an interpretation of P, Q and R. E.g., if $v_P, v_{\neg Q}, v_{\neg R}$ are added, then we are considering the interpretation $P \equiv T, Q \equiv F, R \equiv F$.

The i-th coordinate $i = 1,2,3,4$ corresponds to the i-th conjunct of F. Since $\neg Q$ appears in conjuncts 2 and 4, $v_{\neg Q}$ has a 1 in coordinates 2 and 4, 0 in coordinates 1 and 3. $v_P, v_{\neg P}$ etc. are defined similarly. Now the addition of $v_{\neg Q}$, corresponding to $Q \equiv F$, adds 1 to coordinates 2 and 4 because this interpretation makes the conjuncts 2 and 4 of F TRUE. It should be clear that a vector $\bar{v}$ with

$$(\bar{v})_i = 1 \quad \text{for} \quad i = 5,6,7$$

$$(\bar{v})_i \geq 1 \quad \text{for} \quad i = 1,2,3,4$$

is reachable via V_F iff F is satisfiable.

The vectors v_1, v_2, v_3, v_4 are included to decrease coordinates 1-4 in case an interpretation which satisfies F makes more than one disjunct of a conjunct true. Thus

$v = 1$ is reachable via V_F iff F is satisfiable.

V_F is conflict free.

B. Bounded nets

Let F be as in A. Construct V_F' as follows:

$v_0 = (0,0,0,0,1,1,1)$

$v_1 - v_4$ are as in V_F .

$v_P' = (0,0,0,1,-1,0,0)$; $v_{\neg P}' = (1,1,0,0,-1,0,0)$

$v_Q' = (1,0,1,0,0,-1,0)$; $v_{\neg Q}' = (0,1,0,1,0,-1,0)$

$v_R' = (0,1,0,1,0,0,-1)$; $v_{\neg R}' = (0,0,0,0,0,0,-1)$

The vectors $v_P', \ldots, v_{\neg R}'$ differ from $v_P, \ldots, v_{\neg R}$ of part A only in the last 3 coordinates where -1's replace 1's. Now it is easy to see that $v' = (1,1,1,1,0,0,0)$ is reachable via V_F' iff F is satisfiable. Moreover, V_F' is bounded.

C. Safe nets

The construction involves modifying that of B. to lower the bound to 1. This requires additional coordinates but is straightforward. □

The problem of deciding whether an arbitrary VAS is bounded was shown to be recursive in [7]. The best known upper bound for this problem is Ackermann's function. Safeness is also decidable (in polynomial space). On the other hand, the decidability of persistence is not known through in the next section we do show that if reachability is decidable, then so is persistence. The next theorem shows that all of these properties are PSPACE-hard.

Theorem 2. The following properties of VAS are PSPACE-hard.

A. safeness

B. boundedness

C. persistence

Proof. The proof involves giving a polynomial time bounded reduction of the linear bounded automata (LBA) membership problem to the problem of deciding the various VAS properties. This proves the properties PSPACE-hard using results of [13]. In the following, let $\mathcal{M} = \langle Q, \Sigma, M, q_0, q_F \rangle$ be an LBA where $Q = \{q_1, \ldots, q_r\}$ is the finite set of states; $\Sigma = \{\sigma_1, \ldots, \sigma_s\}$ is the finite tape alphabet $M : Q \times \Sigma \to Q \times \Sigma \times \{R, O, L\}$ the move function (non-deterministic); q_0 is the initial state; q_F is the final state.

$\mathcal{M}$ begins in q_0 on the leftmost symbol of some tape x. Let $|x|$ stand for the length of x. At each step, $\mathcal{M}$ is in a state q scanning the tape symbol σ in location ℓ $(1 \le \ell \le |x|)$. $\mathcal{M}$ then (non-deterministically) shifts to a new state, rewrites σ and moves its read/write head one square right (R), or one square left (L) or does not move its head (O). The tape x is accepted if state q_F is reached.

A. Safeness

Now given $\mathcal{M}$ and x, we build a VAS V such that V is safe if and only x is not accepted by $\mathcal{M}$. Let $n = |x|$.

The VAS will have

$$2(sn+r+n) + 1$$

coordinates. Call the last coordinate c. The tape is coded by n groups of coordinates (one group for each tape position) where each group consists of s coordinates. The i-th coordinate of the j-th group is 1 iff σ_i is the j-th tape symbol. Initially these coordinates contain a coding of x. The next r coordinates give the state, i.e., coordinate i of this group is 1 iff $\mathcal{M}$ is in state q_i. The next n coordinates specify the tape square scanned so that the i-th coordinate of this group will be 1 iff square i is being scanned. For each of the $sn + r + n$ coordinates described above there is a second coordinate whose function is the same. Call the two groups of $sn + r + n$ coordinates, sector 1 and sector 2.

Now v_0 is a coding of x, q_0 and initial square 1, using the first group of $sn + r + n$ coordinates. Coordinate c is initially 0. All other coordinates are initially 0.

For each triple (q, ℓ, σ) consisting of a state q of M, a location ℓ from 1 to n, and a symbol σ in location ℓ, there are 8 vectors $v_i(q, \ell, \sigma) \in V$, $i = 1, \ldots, 8$. There is a different $v_i(q, \ell, \sigma)$ for each combination of sectors where the state, location and symbol information may appear. Each such vector has -1 in one of the sectors in positions corresponding to a) state q; b) location ℓ for the head and; c) symbol σ in the ℓ-th block of coordinates for tape coordinates. It also has +1's in positions in the other sector (i.e., for each of the three pieces of information, in the sector not containing the information) corresponding to the next state, head location and new symbol in the scanned square. Notice that v_i shifts state, location and symbol information from one sector to the other and furthermore the three pieces of information may be in different sectors, thus accounting for the 8 vectors.

If $\mathcal{M}$ reaches q_F, we do not want V to be safe. To do this we have two additional vectors $v, \bar{v}$ in V which when q_F is reached add 2 to coordinate c. These vectors have -1 in the q_F coordinates of sectors 1 and 2 respectively, +2 in coordinate c and 0 in all other coordinates. v or $\bar{v}$ is applicable iff q_F is reached so the VAS is not safe iff $\mathcal{M}$ accepts x.

B. Boundedness

Add two additional coordinates $\bar{c}$, $\bar{\bar{c}}$ to V. These coordinates are intially 0. Add vectors u,w having -1,+1,+1 and +1,-1,+1 in coordinates $c, \bar{c}$ and $\bar{\bar{c}}$ respectively. Then the modified V is unbounded iff q_F is reached iff x is accepted by M.

C. Persistence

Assume that the LBA M is deterministic. Add two additional vectors u,w to V each having -1 in coordinate c and +1 in different coordinates (so $v \neq \bar{v}$). Then V is not persistent iff q_F is reached iff x is accepted by M. □

Corollary 3. The reachability problem for persistent VAS is PSPACE-hard.

Theorem 1 and Corollary 3 give the best known lower bounds for reachability in conflict-free and persistent VAS respectively.

The construction of Theorem 2 also yields Theorem 4. The following are PSPACE-hard where $V = \langle v_0, V\rangle$ is a VAS and c is a coordinate of V. For fixed $k \in N$ is there a $v \in R(V)$ such that $(v)_c = k$? $(v)_c > k$?

4. Persistent VAS

A VAS $V = \langle v_0, V\rangle$ is persistent if for all $v \in R(V)$ and any $v_i, v_j \in V$, $v \xrightarrow[v_i]{}$ and $v \xrightarrow[v_j]{}$ imply $v \xrightarrow[v_i v_j]{}$ $(v_i \neq v_j)$.

In [7], Keller left open the decidability of persistence and its relationship to reachability. A partial answer to these open problems is given by the following theorem.

Theorem 5. Persistence is reducible to reachability.

Proof. To simplify the notation, we restrict our attention to VAS $V = \langle v_0, V\rangle$ where all coordinates of vectors in V are +1,0,-1. The generalization presents no problems. Note that V is not persistent if and only if there is a reachable vector $\bar{v}$ and some $u,w \in V$ such that

$$\circledast \begin{cases} (\bar{v})_i \geq 1 & \text{for all } i \text{ such that } (u)_i = -1 \text{ or } (w)_i = -1 \\ (\bar{v})_i = 1 & \text{for some } i \text{ such that } (u)_i = -1 \text{ and } (w)_i = -1. \end{cases}$$

For each $u,w \in V$ and i such that $(u)_i = (w)_i = -1$, build a VAS V^i_{uw} and check ⊛ using the (supposed) decision procedure for reachability, as follows: add an extra coordinate n + 1 with initial value 1. Let $(v)_{n+1} = 0$ for each $v \in V$. Add to V as follows

a) v' with $(v')_j = -1$ if $(u)_j = -1$ or $(w)_j = -1$

$(v')_{n+1} = -1$

and all other coordinates 0

b) for each $j \neq i$ v'_j with $(v'_j)_j = -1$

all other coordinates 0.

Note that v' can only be applied once. Now 0 is reachable in V^i_{uw} iff v' is applicable to a vector whose i-th coordinate is 1 iff a vector satisfying (*) is reachable in V. □

We have not been able to reduce reachability to persistence though we conjecture that this is possible.

5. Time VAS

In [9], Merlin considered a variant of the Petri net model having a weak timing mechanism. In this section, we show that some of the problems considered in previous sections are undecidable for the time VAS (TVAS) model.

The time vector addition system is a VAS plus a timing mechanism. Associated with each $v \in V$ is a pair of times (t_1, t_2) ($t_1, t_2 \in$ {real numbers} $\cup \{\infty\}$). A vector $v \in V$ is said to be enabled if the TVAS is at a point $u \in N^n$ and $u+v \geq 0$. Assume the $v \in V$ becomes enabled at time t. Then v may not be applied until time $t + t_1$. Moreover, v must be applied by $t + t_2$ (unless it is disabled before then).

We show that the TVAS can be used to simulate input-free 2-counter machines. Since halting is undecidable for input-free 2 counter machines, this will yield the undecidability of various TVAS properties.

An input-free two counter machine is a 6-tuple

$$C = \langle Q, q_0, q_F\ I, C_1, C_2 \rangle$$

where Q is a finite set of states; $q_0 \in Q$ is the initial state; $q_F \in Q$ is the final or halting state; I is a finite set of instructions; and C_1 and C_2 are counters each of which is capable of storing a non-negative integer. The counters are initially set to 0. Instructions are of the form:

a. $(q, D_i, \bar{q})$ $i = 1, 2,$ $q, \bar{q} \in Q$

b. $(q, I_i, \bar{q})$ $i = 1, 2,$ $q, \bar{q} \in Q$

c. $(q, T_i, \bar{q}, \bar{\bar{q}})$ $i = 1, 2,$ $q, \bar{q} \in Q$

The instructions are interpreted as follows

a. $(q, D_i, \bar{q})$: in state q, decrement C_i by 1 and go to $\bar{q}$

b. $(q, I_i, \bar{q})$: in state q, increment C_i by 1 and go to $\bar{q}$

c. $(q, T_i, \bar{q}, \bar{\bar{q}})$: in state q, test C_i; go to $\bar{q}$ if C_i is empty and go to $\bar{\bar{q}}$ otherwise.

Assume the machines are deterministic so that there is at most one instruction for each $q \in Q$. A 2-counter machine halts if it reaches state q_F. (We assume no instruction begins with q_F.) Computations which end by branching to a non-existent state or by attempting to decrement an empty counter are undefined. The following is well known.

Lemma 6. The halting problem for 2-counter machines is undecidable.

Proof. 2-counter machines can simulate Turing machines. □

Our next theorem is proved by showing that TVAS can simulate 2-counter machines.

Theorem 7. The following properties of TVAS are undecidable.

a. reachability

b. boundedness.

Proof. Let $C = \langle Q, q_0, q_F, I, C_1, C_2 \rangle$ be a 2-counter macine. To simplify the notation assume instructions of type a and b satisfy $q \neq \bar{q}$ and those of type c satisfy $q \notin \{\bar{q}, \bar{\bar{q}}\}$. Any 2-counter machine can be modified to satisfy this property without affecting whether or not it halts when started with its counters empty.

The TVAS V which simulates C has one coordinate c_q for each state q of C. The coordinate for $q \in Q$ will be 1 when C is in state q and o otherwise. Initially c_{q_0} is 1 and the other state coordinates are o. There is one coordinate for each counter c_1 for C_1 and c_2 for C_2. Initially c_1 and c_2 are set to o.

Instructions are simulated by vectors in V and associated times. All coordinates not specified are to be 0.

$\underline{I}$	$\underline{V}$	
$(q, D_i, \bar{q})$	$c_q = -1$	$t_1 = 0,\ t_2 = \infty$
	$c_{\bar{q}} = +1$	
	$c_i = -1$	
$(q, I_i, \bar{q})$	$c_q = -1$	$t_1 = 0,\ t_2 = \infty$
	$c_{\bar{q}} = +1$	
	$c_i = +1$	
$(q, T_i, \bar{q}, \bar{\bar{q}})$	$c_q = -1$	$t_1 = 0$
	$c'_q = +1$	$t_2 = 1$
	$c_i = -1$	
	<u>and</u>	
	$c'_q = -1$	$t_1 = 0$
	$c_{\bar{\bar{q}}} = +1$	$t_2 = \infty$
	$c_i = +1$	
	where c'_q is an additional coordinate	
	<u>and</u>	
	$c_q = -1$	$t_1 = 2$
	$c_{\bar{q}} = +1$	$t_2 = 3$

Note that vectors corresponding to instructions which increment or decrement counters can be applied whenever they are enabled. If an instruction tests a counter, there are three vectors associated with it. The first two correspond to the counter not empty case and the third to the counter empty case. If the counter is not empty, the first vector will be applied, disabling the third. This occurs because of

the times associated with the two vectors. The second vector resets the counter to the correct value. If the first vector is not applicable, then the third will eventually be applied. It should be clear that the TVAS *V* simulates the 2-counter machine *C*. Moreover *V* reaches a vector with 1 in coordinate c_{q_F} iff *C* halts. No other vectors are enabled after this occurs ($c_{q_F} = 1$).

a. Reachability - a vector with 1 in coordinate c_{q_F} is reached iff *C* halts. Add vectors to *V* which empty all coordinates if a vector with 1 in c_{q_F} is reached. I.e., add coordinates $\bar{c}$, $\bar{\bar{c}}$ initially 0. Add a vector with $c_{q_F} = -1$, $\bar{c} = +1$. Then add vectors with

1. $\bar{c} = -1$, $\bar{\bar{c}} = +1$, $c_1 = -1$
2. $\bar{c} = -1$, $\bar{\bar{c}} = +1$, $c_2 = -1$
3. $\bar{c} = +1$, $\bar{\bar{c}} = -1$, $c_1 = -1$
4. $\bar{c} = +1$, $\bar{\bar{c}} = -1$, $c_2 = -1$

Then 0 is reached in the modified *V* iff q_F is reached in *C* iff *C* halts.

b. boundedness - Add an additional coordinate $\bar{c}$ which is initially 0. Each vector of *V* adds 1 to $\bar{c}$ so $\bar{c}$ is bounded iff *C* halts.

<u>Conclusion</u>

We have studied reachability for special types of VAS, conflict free and persistent. We believe that it is necessary to more completely understand the mathematical properties of these special cases if the reachability problem for arbitrary VAS is to be tractable.

References

[1] Cook, S. The Complexity of Theorem Proving Procedures, 3-rd Ann. ACM Symp. on Theory of Computing, (1971), 151-158.

[2] Cresp-Reghizzi, S. and Mandrioli, D. A Decidability Theorem for a Class of Vector Addition Systems. Inf. Proc. Letters 3 (1975), 78-80.

[3] Hack, M. The Recursive Equivalence of the Reachability Problem and the Liveness Problem for Petri Nets and Vector Addition Systems. 6 Ann. ACM Symp. on Theory of Comp., May 1974, 303-309.

[4] Holt, A. W. Final Report of the Information System Theory Conference, Tech. Rept. RADC-TR-68-305, Rome Air Dev. Center, Griffiths AFB, N.Y., 1968.

[5] Jones, N. Private Communication

[6] Karp, R. M. Reducibility Among Combinatorial Problems, "Complexity of Computer Computations," ed. R. E. Miller et al, Plenum Press, (1972).

[7] Karp, R. M. and Miller, R. E. Parallel Program Schemata. JCSS 3 (1969), 147-195.

[8] Keller, R.M. Vector Replacement Systems: A Formalism for Modeling Asynchronous Systems. TR 117, Computer Science Lab, Princeton Univ., Dec. 1972.

[9] Merlin, P. M. A Study of the Recoverability of Computing Systems, Ph.D. Thesis - U.C. Irvine (1975).

[10] Stockmeyer, L. J. and Meyer, A. R. Word Problems Requiring Exponential Time, Proc. 5th Ann. ACM Symposium on Theory of Comp. (1973), 1-9.

[11] Petri, C. A. Fundamentals of a Theory of Asynchronous Information Flow, Proc. IFIP Conference, (1962), Munich, 166-168.

[12] VanLeeuwen, J. A Partial Solution to the Reachability Problem for Vector Addition Systems. 6 Ann. ACM Symp. on Theory of Comp., May 1974, 303-309.

[13] Hartmanis, J. and Hunt, H. B. The LBA Problem and its Importance in the Theory of Computing, SIAMAMS Proceedings 7 (1974), 1-25.

[14] Lipton, R. The Reachability Problem and the Boundedness Problem for Petri Nets is Exponential Space Hard, Presented at M.I.T. Conference on Petri Nets and Related Models, Cambridge, July 1975.