Lower Bounds on the State Complexity of Population Protocols*

Philipp Czerner, Javier Esparza

{czerner, esparza}@in.tum.de
Department of Informatics, TU München, Germany

February 24, 2021

Population protocols are a model of computation in which an arbitrary number of indistinguishable finite-state agents interact in pairs. The goal of the agents is to decide by stable consensus whether their initial global configuration satisfies a given property, specified as a predicate on the set of all initial configurations. The state complexity of a predicate is the number of states of a smallest protocol that computes it. Previous work by Blondin *et al.* has shown that the counting predicates $x \geq \eta$ have state complexity $\mathcal{O}(\log \eta)$ for leaderless protocols and $\mathcal{O}(\log \log \eta)$ for protocols with leaders. We obtain the first non-trivial lower bounds: the state complexity of $x \geq \eta$ is $\Omega(\log \log \log \eta)$ for leaderless protocols, and the inverse of a non-elementary function for protocols with leaders.

1. Introduction

Population protocols are a model of computation in which an arbitrary number of indistinguishable finite-state agents interact in pairs to decide if their initial global configuration satisfies a given property. Population protocols were introduced in [5, 6] to study the theoretical properties networks of mobile sensors with very limited computational resources, but they are also very strongly related to chemical reaction networks, a discrete model of chemistry in which agents are molecules that change their states due to collisions.

Population protocols decide a property by *stable consensus*. Each state of an agent is assigned a binary output (yes/no), and in a correct protocol starting at a global

*This work was supported by an ERC Advanced Grant (787367: PaVeS) and by the Research Training Network of the Deutsche Forschungsgemeinschaft (DFG) (378803395: ConVeY).

configuration, all agents eventually reach a consensus by reaching the set of states whose output is the correct answer to the question "does the initial configuration satisfy the property?", and staying in it forever. A typical example of a property decidable by population protocols is majority: initially agents are in one of two initial states, say A and B, and the property to be decided is whether the number of agents in A is larger than the number of agents in B or not. In a seminal paper, Angluin *et al.* showed that population protocols can decide exactly the properties expressible in Presburger arithmetic, the first-order theory of order [9].

In order to define the runtime of a protocol one assumes that at each step a pair of agents is selected uniformly at random and allowed to interact. The *parallel runtime* is then defined as the expected number of interactions until a stable consensus is reached (i.e. until the property is decided), divided by the number of agents. Even though the parallel runtime is computed using a discrete model, under reasonable, commonly accepted assumptions, the result coincides with the runtime of a continuous-time stochastic model. Many papers have investigated the parallel runtime of population protocols, and several landmark results have been obtained. In [6] it was shown that every Presburger property can be decided in $\mathcal{O}(n \log n)$ parallel time, where n is the number of agents, and [8] showed that population protocols with a fixed number of leaders can compute all Presburger predicates in polylogarithmic parallel time. (Loosely speaking, leaders are auxiliary agents that do not form part of the population of "normal" agents, but can interact with them to help them decide the property.) More recent results have studied protocols for majority in which the number of states grows with the number of agents, and shown that polylogarithmic time is achievable by protocols without leaders, even for very slow growth functions, see e.g. [2, 3, 4, 15, 18].

However, many protocols have a high number of states. For example, a quick estimate shows that the fast protocol for majority implicitly described in [8] has tens of thousands of states. This is an obstacle to implementations of protocols in chemistry, where the number of states corresponds to the number of chemical species participating in the reactions. Moreover, the number of states is of fundamental importance because it plays the role of memory in sequential computational models: the total memory available to a protocol is product of the logarithm of the number of states multiplied by the number of agents. Despite these facts, the *state complexity* of a Presburger property, defined as the minimal number of states of any protocol deciding the property, has received comparatively little attention[1]. In [13, 12] Blondin *et al.* have shown that every predicate representable by a boolean combination of threshold and modulo constraints (every Presburger formula can be put into this form) of length n, with numbers encoded in binary, can be decided by a protocol with $p(n)$ states, for some polynomial p. In particular, it is not difficult to see that every property of the form $x \geq \eta$, asking whether the number of agents is at least η, can be decided by a leaderless protocol with $\mathcal{O}(\log \eta)$

[1] Notice that the time-space trade-off results of [2, 3, 4, 15, 18] refer to a more general model in which the number of states of a protocol grows with the number n of agents; in other words, a property is decided by a *family* of protocols, one for each value of n. Trade-off results bound the growth rate needed to compute a predicate within a given time. We study the minimal number of states of a *single* protocol that decides the property for *all* n.

states. A theorem of [12] also proves the existence of an infinite family of thresholds c which has protocols (with leaders) with $\mathcal{O}(\log\log\eta)$ states. However, to the best of our knowledge, there exist no *lower* bounds on the state complexity, i.e. a bound showing that a protocol for $x \geq \eta$ needs $\Omega(f(\eta))$ states for some function f. This question, which was left open in [13], is notoriously hard due to its relation to fundamental questions in the theory of Vector Addition Systems.

In this paper we show that every protocol, with or without leaders, needs a number of states that, roughly speaking, grows like the inverse Ackermann function, and (our main result) that every leaderless protocol for $x \geq \eta$ needs $\Omega(\log\log\log\eta)$ states. The proof of the first bound relies on results on the maximal length of controlled antichains of $\mathbb{N}^d$, a topic in combinatorics with a long tradition in the study of Vector Addition Systems and other models of computation, see e.g. [22, 17, 1, 25, 10]. The triple exponential bound requires us to develop new theory for a generalisation of the antichain condition.

The paper is organised as follows. Section 2 introduces population protocols, state complexity, and its inverse, the busy beaver function for population protocols. Instead of lower bounds on state complexity, for convenience we present upper bounds on the busy beaver function. Section 3 presents some results on the mathematical structure of stable sets of configurations that are used throughout the paper. Section 4 shows an Ackermannian upper bound on the busy beaver function, valid for protocols with or without leaders, and explains why this surprisingly large bound might be optimal. Section 5 gives a triple exponential upper bound on the busy beaver function for leaderless protocols.

2. Population Protocols and State Complexity

2.1. Mathematical preliminaries

For sets A, B we write A^B to denote the set of functions $f : B \to A$. If B is finite we call the elements of $\mathbb{N}^B$ *multisets* over B, and the elements of $\mathbb{R}^B$ vectors of dimension $|B|$. Arithmetic operations on vectors in $\mathbb{R}^B$ are defined as usual, extending the vectors with zeroes if necessary. For example, if $B' \subseteq B$, $x \in \mathbb{R}^B$ and $y \in \mathbb{R}^{B'}$ then $x + y \in \mathbb{N}^B$ is defined by $(x+y)_b = x_b + y_b$, where $y_b = 0$ for every $b \in B \setminus B'$. For $x, y \in \mathbb{R}^B$ we write $x \leq y$ if $x_i \leq y_i$ for all $i \in B$, and $x \lneq y$ if $x \leq y$ and $x \neq y$. Abusing language we identify an element $b \in B$ with the one-element multiset containing it, i.e. $x \in \mathbb{N}^B$ with $x_b = 1$ and $x_i = 0$ for $i \neq b$. We also write $|x| := \sum_{b \in B} x_b$ for the total number of elements in a multiset $x \in \mathbb{N}^B$, and $\mathbf{1}$ to denote the all-ones vector of appropriate dimension. Finally, given a vector $v \in \mathbb{R}^k$, we define $\|v\|_1 = \sum_{i=1}^k |v_i|$ and $\|v\|_\infty = \max_{i=1}^k |v_i|$

2.2. Population protocols

We recall the population protocol model of [5, 7], with explicit mention of leader agents. A *population protocol* is a tuple $\mathcal{P} = (Q, T, L, X, I, O)$ where Q is a finite set of *states*; $T \subset Q^2 \times Q^2$ is a set of *transitions*; $L \in \mathbb{N}^Q$ is the *leader multiset*; X is a finite set of *input variables*; $I : X \to Q$ is the *input mapping*; and $O : Q \to \{0,1\}$ is the *output mapping.*

Inputs and configurations An *input* is a multiset $v \in \mathbb{N}^X$ such that $|v| \geq 2$, and a *configuration* is a multiset $m \in \mathbb{N}^Q$ such that $|m| \geq 2$. Intuitively, a configuration represents a population of agents where $m(q)$ denotes the number of agents in state q. The *initial configuration* for input v is defined as $IC(v) := L + \sum_{x \in X} v(x) \cdot I(x)$. Abusing language, throughout the paper we write $IC(i)$ instead of $IC(i \cdot x)$ to denote the initial configuration for input $i \in \mathbb{N}$, if $\mathcal{P}$ has a unique input state $\{x\} = X$.

The *output* $O(m)$ of a configuration m is b if $m(q) \geq 1$ implies $O(q) = b$ for all $q \in Q$, and undefined otherwise. So a population has output b if all agents have output b.

Executions A transition $t = ((p,q),(p',q'))$ is *enabled* in a configuration m if $m \geq p + q$, and *disabled* otherwise. As $|m| \geq 2$, every configuration enables at least one transition. If t is enabled in m, then it can be *fired* leading to configuration $v := m - p - q + p' + q'$, which we denote $m \xrightarrow{t} v$. Given a sequence $\sigma = t_1 t_2 ... t_n$ of transitions, we write $m \xrightarrow{\sigma} v$ if there exist configurations $m_1, m_2, ..., m_n$ such that $m \xrightarrow{t_1} m_1 \xrightarrow{t_2} m_2 \cdots m_n \xrightarrow{t_n} m'$, and $m \xrightarrow{*} m'$ if $m \xrightarrow{\sigma} m'$ for some sequence $\sigma \in T^*$. For every set of transitions $T' \subseteq T$, we write $m \xrightarrow{T'} m'$ if $m \xrightarrow{t} m'$ for some $t \in T'$, and $m \xrightarrow{T'^*} m'$ if $m \xrightarrow{\sigma} m'$ for some sequence $\sigma \in T'^*$. Given a set M of configurations, $m \xrightarrow{*} M$ denotes that $m \xrightarrow{*} m'$ for some $m' \in M$.

An *execution* is a sequence of configurations $\sigma = m_0 m_1 ...$ such that $m_i \to m_{i+1}$ for every $i \in \mathbb{N}$. The *output* $O(\sigma)$ of σ is b if there exist $i \in \mathbb{N}$ such that $O(m_i) = O(m_{i+1}) = ... = b$, otherwise $O(\sigma)$ is undefined.

Executions have the *monotonicity property*: If $m_0 m_1 m_2 ...$ is an execution, then for every configuration D the sequence $(m_0 + m)(m_1 + m)(m_2 + m)...$ is an execution too. We often say that a statement holds "by monotonicity" , meaning that it is a consequence of the monotonicity property.

Computations An execution $\sigma = m_0 m_1 ...$ is *fair* if for every configuration m the following holds:

$$\text{if } |\{i \in \mathbb{N} : m_i \xrightarrow{*} m\}| \text{ is infinite, then } |\{i \in \mathbb{N} : m_i = m\}| \text{ is infinite.}$$

In other words, fairness ensures that an execution cannot avoid a configuration forever. We say that a population protocol *computes* a predicate $\varphi : \mathbb{N}^X \to \{0,1\}$ (or *decides* the property represented by the predicate) if for every $v \in \mathbb{N}^X$ every fair execution σ starting from $IC(v)$ satisfies $O(\sigma) = \varphi(v)$. Two protocols are *equivalent* if they compute the same predicate. It is known that population protocols compute precisely the Presburger-definable predicates [9].

Example 1. *Let $\mathcal{P}_n = (Q, T, 0, \{x\}, I, O)$ be the protocol where $Q := \{0, 1, 2, 3, ..., 2^n\}$, $I(x) := 1$, $O(a) = 1$ iff $a = 2^n$, and for each $a, b \in Q$ the set T of transitions contains $((a,b),(0,a+b))$ if $a + b < 2^n$, and $((a,b),(2^n,2^n))$ if $a + b \geq 2^n$. It is readily seen that*

$\mathcal{P}_n$ computes $x \geq 2^n$ with 2^n+1 states. Intuitively, each agent stores a number, initially 1. When two agents meet, one of them stores the sum of their values and the other one stores 0, with sums capping at 2^n. Once an agent reaches this cap, all agents eventually get converted to 2^n.

Now, consider the protocol $\mathcal{P}'_n = (Q', T', 0, \{x\}, I', O')$, where $Q' := \{0, 2^0, 2^1, ..., 2^n\}$, $I'(x) := 2^0$, $O'(a) = 1$ iff $a = 2^n$, and T' contains $((2^i, 2^i), (0, 2^{i+1}))$ for each $0 \leq i < n$, and $((a, 2^n), (2^n, 2^n))$. for each $a \in Q'$ It is easy to see that $\mathcal{P}'_n$ also computes $x \geq 2^n$, but more succinctly: While $\mathcal{P}_n$ has 2^n+1 states, $\mathcal{P}'_n$ has only $n+1$ states.

Leaderless protocols A protocol $\mathcal{P} = (Q, T, L, X, I, O)$ is *leaderless* if $L = \emptyset$, and *has $|L|$ leaders* otherwise. Protocols with leaders and leaderless protocols compute the same predicates [9]. For $L = \emptyset$ we have

$$\begin{aligned}
\lambda IC(v) + \lambda' IC(v') &= \lambda\Big(L + \sum_{x \in X} v(x) \cdot I(x)\Big) + \lambda'\Big(L + \sum_{x \in X} v'(x) \cdot I(x)\Big) \\
&= \lambda \sum_{x \in X} v(x) \cdot I(x) + \lambda' \sum_{x \in X} v'(x) \cdot I(x) \\
&= IC(\lambda v + \lambda' v')
\end{aligned}$$

for all inputs v, v' and $\lambda, \lambda' \in \mathbb{N}$. In other words, any linear combination of configurations with natural coefficients is also an initial configuration.

2.3. State complexity of population protocols

Informally, the state complexity of a predicate is the minimal number of states of the protocols that compute it. Given n, we would like to define the function $STATE(n)$ as the maximum state complexity of the predicates of size at most n. However, defining the size of a predicate requires to fix a representation. Population protocols compute exactly the predicates expressible in Presburger arithmetic [9], and so there are at least three natural representations: formulas of Presburger arithmetic, existential formulas of Presburger arithmetic, and semilinear sets [19]. However, the translations between these representations involve superexponential blow-ups. For this reason we focus on threshold predicates of the form $x \geq \eta$, for which the size of the predicate is just the size of η, independently of whether the predicate is described as a formula or a semilinear set. We choose to encode numbers in unary, and so we define $STATE(n)$ as the number of states of the smallest predicate computing $x \geq \eta$.

The inverse of $STATE(n)$ is the function that assigns to a number n the largest η such that a protocol with n states computes $x \geq \eta$. Recall that the busy beaver function assigns to a number n the largest η such that a Turing machine with n states started on a blank tape writes η consecutive ones on the tape and terminates. Due to this analogy, we call the inverse of $STATE$ the *busy beaver function* for population protocols, and call protocols computing predicates of the form $x \geq \eta$ busy beaver protocols, or just *busy beavers*.

Definition 1. *The* busy beaver function $BB : \mathbb{N} \to \mathbb{N}$ *is defined as follows: $BB(n)$ is the largest $\eta \in \mathbb{N}$ such that the predicate $x \geq \eta$ is computed by some leaderless protocol with at most n states. The function $BB_L(n)$ is defined analogously, but for general protocols, possibly with leaders.*

In [13] Blondin *et al.* give lower bounds on the busy beaver function:

Theorem 2 ([13])**.** *$BB(n) \in \Omega(2^n)$ and $BB_L(n) \in \Omega(2^{2^n})$.*

However, to the best of our knowledge no upper bounds have been given.

3. Mathematical Structure of Stable Sets

A set M of configurations is *downward closed* if $m \in M$ and $m' \leq m$ implies $m' \in M$. A pair (μ, S), where μ is a configuration and $S \subseteq Q$, is a *base element* of M if $\mu + \mathbb{N}^S \subseteq M$. A *base* of M is a finite set $\mathcal{B}$ of base elements such that $M = \bigcup_{(\mu,S)\in\mathcal{B}}(\mu + \mathbb{N}^S)$. It is well-known (an easy consequence of Dickson's lemma), that every downward-closed set of configurations has a base. We define the *norm* of a base element (μ, S) as $\|(\mu, S)\|_\infty := \|\mu\|_\infty$, and the norm of a base as the maximal norm of its elements. We apply these notions to the *stable configurations* of the protocol:

Definition 2. *Let $b \in \{0, 1\}$. A configuration m is b-*stable *if $O(m') = b$ for every configuration m' reachable from m. The set of b-stable configurations is denoted SC_b.*

It follows easily from the definitions that a population protocol *computes* a predicate $\varphi : \mathbb{N}^X \to \{0, 1\}$ iff $IC(v) \xrightarrow{*} SC_0$ for every input v satisfying $\varphi(v) = 0$, and $IC(v) \xrightarrow{*} SC_1$ for every input v satisfying $\varphi(v) = 1$.

Lemma 3. *Let $\mathcal{P}$ be a protocol with n states. For every $b \in \{0, 1\}$ the set SC_b is downward closed and has a base of norm at most $2^{2(2n+1)!+1}$. In particular, SC_b has a base with at most $\vartheta(n) := 2^{(2n+2)!}$ elements.*

Proof. We first show that SC_b is downward closed, by showing that its complement $\overline{SC_b}$ is upward closed. Assume $m \in \overline{SC_b}$ and $m' \geq m$. We prove $m' \in \overline{SC_b}$. Since $m \in \overline{SC_b}$ we have $m \xrightarrow{*} m''$ for some m'' such that $O(m'') \neq b$. By monotonicity, $m' = m+(m'-m) \xrightarrow{*} m''+(m'-m)$, and since $O(m'') \neq b$ we have $O(m''+(m'-m)) \neq b$. So $m' \in \overline{SC_b}$.

For the second part, let $\beta := 2^{2(2n+1)!}$ and fix a b-stable configuration m. Let $S := \{q \in Q : m_q > 2\beta\}$, and define $\mu \leq m$ as follows: $\mu_i := m_i$ for $i \notin S$ and $\mu_i := 2\beta$ for $i \in S$. Since $\mu \leq m$ and m is b-stable, so is μ. We show that (μ, S) is a base element of SC_b, which proves the result. Assume the contrary. Then some configuration $m' \in \mu + \mathbb{N}^S$ is not b-stable. So $m' \xrightarrow{*} m''$ for some m'' satisfying $m''(q) \geq 1$ for some state $q \in Q$ with $O(q) \neq b$; we say that m'' covers q. By Rackoff's Theorem [24], m'' can be chosen so that $m' \xrightarrow{\sigma} m''$ for a sequence σ of length $2^{2^{\mathcal{O}(n)}}$; a more precise bound is $|\sigma| \leq \beta$ (see [16, Theorem 3.12.11]). Since a transition moves at most two agents out of a given state, σ moves at most 2β agents out of a state. So, by the definition of μ, the sequence σ is also executable from μ, and also leads to a configuration that covers q. But this contradicts that μ is b-stable.

To prove the bound on the number of elements of the base, observe that the number of pairs (μ, S) such that μ has norm at most k and $S \subseteq Q$ is at most $(k+2)^n$. Indeed, for each state q there are at most $k+2$ possibilities: $q \in S$, or $q \notin S$ and $0 \leq \mu(q) \leq k$. So $\vartheta \leq (2^{2(2n+1)!+1} + 2)^n \leq 2^{(2n+2)!}$. □

From now on we use the following terminology:

Definition 3. *A* b-base *is a base of* SC_b *of norm at most* $2^{2(2n+1)!+1}$*, and its elements are called* b-base elements*.*

4. A General Upper Bound on the Busy Beaver Function

Our general strategy to find upper bounds for the busy-beaver function $BB_L(n)$ is as follows:

(1) Prove a "Pumping Lemma" stating that if a protocol rejects two inputs $a < b$ satisfying certain conditions, then it rejects all inputs of the form $a + \lambda(b - a)$, for every $\lambda \geq 0$, and so that it does not compute the predicate $x \geq \eta$.

(2) Using the Pumping Lemma, we reduce the existence of the inputs a and b to the existence of a finite sequence of vectors of dimension n satisfying certain purely combinatorial properties. Moreover, the size of b is linked to the length of the sequence.

(3) Say a sequence satisfying the properties of (2) is *bad* (it implies that the protocol does not compute $x \geq \eta$), and otherwise *good*. We provide a bound $B(n)$ on the maximal length of good sequences.

It follows from (1)-(3) that a protocol with n states cannot compute $x \geq \eta$ for any $\eta \geq B(n)$. Indeed, if $\eta \geq B(n)$ then every sequence of vectors of dimension n and length η is bad. So the sequence satisfies the conditions of the Pumping Lemma, and so the protocol rejects all inputs of the form $a + \lambda(b - a)$. In this section we follow this strategy to provide an upper bound valid for all protocols. In the next section we apply it again, albeit in a more sophisticated way, to obtain a far better upper bound for leaderless protocols.

Fix a protocol $\mathcal{P}_n = (Q, T, 0, \{x\}, I, O)$ with $|Q| = n$. We start by stating and proving the Pumping Lemma.

Lemma 4 (Pumping Lemma)**.** *If there exist inputs a and b, a* 0-*base element* (μ, S)*, and configurations* $m_a, m_b \in \mu + \mathbb{N}^Q$ *satisfying (1)* $m_a \leq m_b$*, (2)* $IC(a) \xrightarrow{*} m_a$*, and (3)* $m_a + IC(b - a) \xrightarrow{*} m_b$*, then* $\mathcal{P}$ *rejects* $a + \lambda(b - a)$ *for every* $\lambda \geq 0$*.*

Proof. We first claim that $m_a + IC(\lambda(b-a)) \xrightarrow{*} m_a + \lambda(m_b - m_a)$ holds for every $\lambda \geq 0$ $(*)$. The proof is by induction on λ. The basis $\lambda = 0$ is trivial. For the induction step let $\lambda \geq 1$. Due to monotonicity, we have

$$\begin{array}{rcll} IC(a+\lambda(b-a)) & = & IC(a) + IC(b-a) + IC((\lambda-1)(b-a)) & \\ & \xrightarrow{*} & m_a + IC(b-a) + IC((\lambda-1)(b-a)) & (2) \\ & \xrightarrow{*} & m_a + (m_b - m_a) + IC((\lambda-1)(b-a)) & (3) \\ & \xrightarrow{*} & m_b + (m_b - m_a) + (\lambda-1)(m_b - m_a) & (*) \\ & = & m_a + \lambda(m_b - m_a) & \end{array}$$

and the claim is proved. By (1) and $m_a, m_b \in \mu + \mathbb{N}^S$ we have $m_a + \lambda(m_b - m_a) \in \mu + \mathbb{N}^S$ for every $\lambda \geq 0$. So, by (2) and the claim, SC_0 is reachable from $IC(a + \lambda(b-a))$ for every $\lambda \geq 0$. So $\mathcal{P}$ rejects 0 for every input $a + \lambda(b-a)$. $\square$

Our goal now is to find a bound $B(n)$ such that for every protocol with at most n states there are inputs $a < b \leq B(n)$ rejected by the protocol and satisfying conditions (1)-(3) of the Pumping Lemma. For this, observe that for every rejected input i we have $IC(i) \xrightarrow{*} m_i$ for some configuration $m_i \in SC_0$, and so $m_i \in \mu_i + \mathbb{N}^{S_i}$ for some 0-base element (μ_i, S_i). Further, by Lemma 3 we can assume that the 0-base has $\vartheta(n)$ elements. The triple (μ_i, S_i, m_i) can be seen as a "rejection certificate" for i. The certificate can be verified by checking that $m_i \in \mu_i + \mathbb{N}^{S_i}$, and finding σ such that $IC(i) \xrightarrow{\sigma} m_i$.

Definition 4. *For every rejected input* $2 \leq i \leq \eta - 1$*, the triple* $cert(i) := (\mu_i, S_i, m_i)$ *is the* (rejection) certificate *of* i*. We call* (μ_i, S_i) *the* type *of* $cert(i)$*. The* certificate sequence *of* $\mathcal{P}$ *is the sequence* $cert(2)cert(3)...cert(\eta - 1)$*.*

The conditions of the Pumping Lemma can now be reformulated as follows: there are two inputs a, b such that their certificates have the same type and satisfy condition (1). Since there are at most $\vartheta(n)$ types, if the number of rejected inputs exceeds $\vartheta(n)$, then there are two inputs a, b such that $cert(a)$ and $cert(b)$ have the same type. However, their certificates need not yet satisfy condition (1). To solve this problem we examine the certificate sequence in more detail. More precisely, we examine the sequence $m_2\, m_3 \cdots m_{\eta-1}$. Using the terminology of [17], it is a *linearly controlled sequence*, meaning that there is a linear control function $f : \mathbb{N} \to \mathbb{N}$ satisfying $|m_i| \leq f(i)$. Indeed, since $IC(i) \xrightarrow{*} m_i$, we have $|m_i| = |IC(i)| = |L| + i$, and so we can take $f(n) = |L| + n$. This allows us to use a result on linearly controlled sequences from [17]. Say a finite sequence $v_0, v_1, \cdots, v_s$ of vectors of the same dimension is *bad* if there are two indices $0 \leq i_1 < i_2 \leq s$ such that $v_{i_1} \leq v_{i_2}$.[2] Dickson's Lemma shows that every infinite sequence sequence of vectors contains a bad prefix, and the result extends from two to three or any other finite number of indices $i_1 < i_2 < ....$

The maximal length of good linearly controlled sequences has been studied in [22, 17, 10], and the results been used to bound the runtime of algorithms to check properties of a number of computational models, including Vector Addition Systems, and Counter Machines [21]. The following lemma follows easily from results of [17]:

[2]Unfortunately, in [17] bad sequences are called "good". We risk this confusion to maintain the convention that a bad sequence is a witness of the fact that the protocol does *not* compute $x \geq \eta$.

Lemma 5. *[17] For every $\delta \in \mathbb{N}$ and for every elementary function $g : \mathbb{N} \to \mathbb{N}$, there exists a function $F_{\delta,g} : \mathbb{N} \to \mathbb{N}$ at level $\mathcal{F}_\omega$ of the Fast Growing Hierarchy satisfying the following property: For every infinite sequence $v_0, v_1, v_2 \ldots$ of vectors of $\mathbb{N}^n$ satisfying $|v_i| \leq i + \delta$, there exist $i_0 < i_1 < \ldots < i_{g(n)} \leq F_{\delta,g}(n)$ such that $v_{i_0} \leq v_{i_1} \leq \cdots \leq v_{g(n)}$.*

For the definition of the Fast Growing Hierarchy see [17]. For our purposes it suffices to know that $\mathcal{F}_\omega$ contains functions that, crudely speaking, grow like the Ackermann function. Using the lemma we obtain:

Theorem 6. *Let $\mathcal{P}$ be a population protocol with n states and ℓ leaders computing a predicate $x \geq \eta$ for some $\eta \geq 2$. Then $\eta < F_{\ell,\vartheta}(n)$, where $\vartheta(n)$ is the function of Lemma 3.*

Proof. We inductively define a sequence $m_2, m_3, m_4 \ldots$ of configurations of $SC_0 \cup SC_1$ satisfying:

(i) $IC(i) \xrightarrow{*} m_i$ for every $n \geq 2$.

(ii) $m_i + (j - i) \cdot I(x) \xrightarrow{*} m_j$ for every $2 \leq i \leq j$. (Observe that $m_i + (j - i) \cdot I(x)$ is the result of adding $(j - i)$ agents in state $I(x)$ to m_i.)

Since $\mathcal{P}$ computes $x \geq \eta$, for every $i \geq 2$ every fair run of $\mathcal{P}$ starting at $IC(i)$ eventually reaches SC_0 or SC_1 (depending on whether $i < \eta$ or $i \geq \eta$), and stays there forever. We define $m_2, m_3, m_4, \ldots$. First, we let m_2 be any configuration of $SC_0 \cup SC_1$ reachable from $IC(2)$. Then, for every $i \geq 2$, assume that m_i has already been defined and satisfies $IC(i) \xrightarrow{*} m_i$. Observe that $IC(i+1) = IC(i) + I(x)$. Since $IC(i) \xrightarrow{*} m_i$, we also have $IC(i+1) = IC(i) + I(x) \xrightarrow{*} m_i + I(x)$. This execution can be extended to a fair run, which eventually reaches $SC_0 \cup SC_1$. Let m_{i+1} be any configuration of $SC_0 \cup SC_1$ reachable from $m_i + I(x)$.

Let us show that $m_2, m_3, m_4 \ldots$ satisfies (i) and (ii). Property (i) holds for m_2 by definition, and for $i \geq 2$ because $IC(i+1) = IC(i) + I(x) \xrightarrow{*} m_i + I(x) \xrightarrow{*} m_{i+1}$. For property (ii), by monotonicity and the definition of m_i we have for every $2 \leq i \leq j$:

$$m_i + (j - i) \cdot I(x) \xrightarrow{*} m_{i+1} + (j - i - 1) \cdot I(x) \xrightarrow{*} \cdots \xrightarrow{*} m_{j-1} + I(x) \xrightarrow{*} m_j$$

Assume $\eta > F_{\ell,\vartheta}(n)$. By Lemma 5 there exist $\vartheta(n) + 1$ indices $i_0 < i_1 < \ldots < i_{\vartheta(n)} \leq F_{\ell,\vartheta}(n)$ such that $m_{i_0} \leq m_{i_1} \leq \cdots \leq m_{i_{\vartheta(n)}}$. Since $\mathcal{P}$ computes $x \geq \eta$, every m_{i_j} belongs to SC_0. By the definition of ϑ and the pigeonhole principle, there are $a < b$ and a 0-base element (μ, S) such that $m_{i_a}, m_{i_b} \in \mu + \mathbb{N}^S$. Rename $a := i_a$ and $b := i_b$. By property (ii) we have $m_a + (b - a) \cdot I(x) \xrightarrow{*} m_b$. Since $m_a, m_b \in \mu + \mathbb{N}^S$, applying Lemma 4 we get that $\mathcal{P}$ rejects $a + \lambda(b - a)$ for every $\lambda \geq 0$. This contradicts that $\mathcal{P}$ computes $x \geq \eta$. □

4.1. Is the bound optimal?

The function $F_{\ell,\vartheta}(n)$ grows so fast that one can doubt that the bound is even remotely close to optimal. However, recent results show that this would be less strange than it seems. If a protocol $\mathcal{P}$ computes a predicate $x \geq \eta$, then η is the smallest number such that $IC(\eta+1) \xrightarrow{*} SC_1$. Therefore, letting $\mathbf{BBP}(n)$ denote the busy beaver protocols with at most n states, and letting $SC_1^{\mathcal{P}}$ and $IC^{\mathcal{P}}$ denote the set SC_1 and the initial mapping of the protocol $\mathcal{P}$, we obtain:

$$BB_L(n) = \max_{\mathcal{P} \in \mathbf{BBP}(n)} \min\{i \in \mathbb{N} \mid \exists m \in SC_1^{\mathcal{P}} : IC^{\mathcal{P}}(i) \xrightarrow{*} m\}$$

Consider now two deceptively similar functions. Let All_1 be the set of configurations m such that $O(m) = 1$, i.e. all agents are in states with ouput 1. Further, let $Some_1$ be the set of configurations m such that $O(m) \neq 0$, i.e. at least one agent is in a state with output 1. Finally, let $\mathbf{PP}(n)$ denote the set of *all* protocols with alphabet $X = \{x\}$, possibly with leaders, and n states. Notice that we include also the protocols that do not compute any predicate. Define

$$\begin{aligned} f_1(n) &= \max_{\mathcal{P} \in \mathbf{PP}(n)} \min\{i \in \mathbb{N} \mid \exists m \in All_1^{\mathcal{P}} : IC^{\mathcal{P}}(i) \xrightarrow{*} m\} \\ f_2(n) &= \max_{\mathcal{P} \in \mathbf{PP}(n)} \min\{i \in \mathbb{N} \mid \exists m \in Some_1^{\mathcal{P}} : IC^{\mathcal{P}}(i) \xrightarrow{*} m\} \end{aligned}$$

Using recent results in Petri nets and Vector Addition Systems [14, 20] it is easy to prove that $f_1(n)$ grows faster than any elementary function[3]. However, a recent result [11] by Balasubramanian *et al.* shows $f_1(n) \in 2^{\mathcal{O}(n)}$ for leaderless protocols! Finally, we get $f_2(n) \in 2^{2^{\mathcal{O}(n)}}$ from a classical result on the coverability problem for Vector Addition Systems [24] due to Rackoff[4].

These results show that a non-elementary bound on $BB_L(n)$ might well be optimal. However, we now prove that this can only hold for population protocols with leaders. We show $BB(n) \in 2^{2^{2^{\mathcal{O}(n)}}}$, i.e. leaderless busy beavers with n states can only compute predicates $x \geq \eta$ for numbers η at most triple exponential on n.

The scheme of the proof is similar to that of Theorem 6. In particular, we also rely on a lemma bounding the length of certain good sequences of vectors. However, the definition of a good sequence is a different one, which to the best of our knowledge has not been studied yet. Therefore, instead of resorting to [17] we develop the machinery to prove the lemma by ourselves.

[3] The paper [20] considers protocols with one leader, and studies the problem of moving from a configuration with the leader in a state q_{in} and all other agents in another state r_{in}, to a configuration with the leader in a state q_f and all other agents in state r_f. The paper uses results from [14] to show that the smallest number of agents for which this is possible grows faster than any elementary function in the number of states of the protocol.

[4] The problem reduces to reaching a configuration with at least one agent in states of output 1. Rackoff shows that, if this is possible for some n, then the configuration can be reached in at most $2^{2^{\mathcal{O}(n)}}$ steps, which can only involve $2^{2^{\mathcal{O}(n)}}$ agents.

5. An Upper Bound for Leaderless Protocols

The main property of leaderless protocols, already mentioned in Section 2, is that, loosely speaking, initial configurations are closed under linear combinations:

$$\lambda IC(i) + \lambda' IC(j) = IC(\lambda i + \lambda' j) \qquad \text{for every } i, j, \lambda, \lambda' \in \mathbb{N}$$

This extends to executions: If $IC(i) \xrightarrow{*} m_i$ and $IC(j) \xrightarrow{*} m_j$, then $IC(\lambda i + \lambda' j) \xrightarrow{*} \lambda m_i + \lambda' m_j$. We make use of this property throughout the section without explicit mention. Observe also that even if the coefficients of the linear combination are nonnegative rationals we can always multiply them by a suitable constant and obtain an execution.

We reuse the proof strategy described at the beginning of Section 4. In Section 5.1 we state and prove a Pumping Lemma showing that under certain conditions the protocol rejects infinitely many inputs, contradicting that it computes $x \geq \eta$. In Section 5.2 we introduce QCT-sequences of vectors, and show that a bound on the length of the so called good QCT-sequences implies a bound on η. Finally, Section 5.3 derives a bound on the length of good QCT-sequences.

Notation. *For the rest of the section we fix a leaderless population protocol $\mathcal{P} = (Q, T, \emptyset, \{x\}, I, O)$ with n states.*

5.1. A Pumping Lemma

We first introduce some preliminaries, then formulate a first version of the Pumping Lemma (Lemma 7), and then strengthen it, yielding the final version (Lemma 10).

Potential reachability and a first Pumping Lemma.

Let $t = ((p, q), (p', q'))$ be a transition. By the definition of the reachability relation, for every two configurations m, m', if $m \xrightarrow{t} m'$ then $m' = m - p - q + p' + q'$. This motivates the following definition:

Definition 5. *Let $t = ((p, q), (p', q'))$ be a transition. The* effect *of t is the vector $\Delta(t) := q' + r' - q - r \in \mathbb{Z}^Q$. Given a sequence $\sigma = t_1 ... t_n$, we denote its* Parikh vector *$\overrightarrow{\sigma} := \sum_{t \in \sigma} t \in \mathbb{N}^T$ as the vector mapping each transition to its number of occurrences in σ. We can then define the* effect *of σ as $\Delta(\sigma) := \sum_{t \in T} (\overrightarrow{\sigma})_t \Delta(t) = \sum_{t \in \sigma} \Delta(t)$.*

Observe that, since a transition only involves two agents, all components of $\Delta(t)$ lie in the interval $[-2, 2]$. Further, observe that $m \xrightarrow{\sigma} m'$ implies $m' = m + \Delta(\sigma)$. In particular, m' depends only on the effect of σ. We introduce some useful notations:

Definition 6. *For every sequence $\sigma \in T^*$ of transitions, we write $m \overset{\sigma}{\Longrightarrow} m'$ if $m' = m + \Delta(\sigma)$. Further, we write $m \overset{*}{\Longrightarrow} m'$ if there exists σ such that $m \overset{\sigma}{\Longrightarrow} m'$, and say that m' is* potentially reachable *from m.*

We make the following observations:

- If $m \xrightarrow{\sigma} m'$ then $m \xRightarrow{\sigma} m'$, but the converse does not hold in general.
- For fixed configurations m, m', whether $m \xrightarrow{\sigma} m'$ holds or not depends only on the Parikh vector $\overrightarrow{\sigma}$.
- If $m \xRightarrow{\sigma} m'$ and $m \geq 2|\sigma|$, then $m \xrightarrow{\sigma} m'$. This follows immediately from the fact that each transition moves exactly two agents.

We formulate a first version of a pumping lemma, which we will then strengthen.

Lemma 7. *If there is $\gamma \in \mathbb{N}$ and inputs $s_0, a, b \in \mathbb{N}$ such that*

(1) $IC(s_0) \xrightarrow{*} m$ for some $m \geq 2\gamma$,

(2) $m + IC(a) \xrightarrow{*} \mu + \mathbb{N}^S$ for some 0-base element (μ, S), and

(3) $IC(b) \xRightarrow{\sigma} \mathbb{N}^S$ for some σ such that $|\sigma| \leq \gamma$,

then $\mathcal{P}$ rejects $(s_0 + a + \lambda b)$ for every $\lambda \geq 0$.

Proof. By (2) and (3) there exist configurations $v, u \in \mathbb{N}^S$ s.t. $m + IC(a) \xrightarrow{*} \mu + v$ and $IC(b) \xRightarrow{\sigma} u$. Since a transition removes at most two agents from a state, and we have $m \geq 2\gamma$ and $|\sigma| \leq \gamma$, the sequence σ is enabled at m, and so also at $m + IC(b)$. By (3) we obtain $m + IC(b) \xrightarrow{\sigma} m + u$. So we get

$$\begin{array}{rcll} IC(s_0 + a + \lambda b) & = & IC(s_0) + IC(a) + \lambda IC(b) & \\ & \xrightarrow{*} & m + IC(a) + \lambda IC(b) & \\ & \xrightarrow{*} & m + IC(a) + (\lambda - 1) IC(b) + u & \\ & \xrightarrow{*} & \cdots \xrightarrow{*} m + IC(a) + \lambda u & (1) \\ & \xrightarrow{*} & \mu + v + \lambda u & (2) \end{array}$$

Since (μ, S) is a 0-base element and $v, u \in \mathbb{N}^S$, we have $\mu + v + \lambda u \in SC_0$, and so $\mathcal{P}$ rejects $(s_0 + a + \lambda b)$. □

A stronger pumping lemma.

In the rest of the section we show that Lemma 7 can be strengthened by fixing the values of γ and s_0, that is, one only needs to look for inputs i and j in order to "pump". We first show that the sequence σ can always be chosen of size at most $(n+1)^6 4^n$.

Lemma 8. *If $IC(i) \xRightarrow{*} \mathbb{N}^S$ for some input $i \geq 2$ then $IC(j) \xRightarrow{\sigma} \mathbb{N}^S$ for some input $j \geq 2$ and a sequence σ of length at most $(n+1)^6 4^n$.*

Proof. We construct a linear system of inequalities, any integer solution of which will yield a desired sequence σ with $IC(j) \xRightarrow{\sigma} \mathbb{N}^S$ for some j. Then we apply a well-known bound, showing that a small solution exists.

To construct this system, we use what is known in the analysis of Petri nets as the marking equation (see e.g. [23]). Since the order of transitions in σ does not matter, we consider the Parikh vector $\overrightarrow{\sigma} = \sum_{t\in\sigma} t \in \mathbb{N}^T$ of σ, as defined above. Defining $\mathcal{A} : Q \times T \to \mathbb{Z}$ as the matrix where the t-th column is precisely $\Delta(t)$ for $t \in T$, we get that the effect of the whole sequence is simply $\Delta(\sigma) = \mathcal{A}\overrightarrow{\sigma}$.

Therefore the statement $IC(j) \overset{\sigma}{\Longrightarrow} \mathbb{N}^S$ is equivalent to the following system of linear inequalities over the vector u of variables, where $v_i := (\mathcal{A}_{it})_{t\in T}$ denotes the i-th row of $\mathcal{A}$, for $i \in Q$, and $x \in Q$ the unique initial state of $\mathcal{P}$:

$$\begin{aligned} \exists u \in \mathbb{N}^T : v_x^\top u &\leq -1 & \\ v_i^\top u &= 0 & \text{for all } i \in Q \setminus S_c, i \neq x \\ v_i^\top u &\geq 0 & \text{for every } i \in S_c, i \neq x \end{aligned}$$

We know that the above system has a solution for u, so it also has an integer solution with coefficients at most $\gamma = (n^2 + 1)4^n$. This bound follows from a result by von zur Gathen and Sieveking [27], combined with the estimate that for any submatrix of $\mathcal{A}$ its determinant has absolute value at most 4^n. The bounds on the determinants follows directly from a suitable Laplace expansion of the columns, as each transition $t \in T$ has $\|\Delta(t)\|_1 \leq 4$. Since there are $|T| \leq n^4$ different transitions, the total number of transitions in σ is at most $n^4(n^2 + 1)4^n \leq (n + 1)^6 4^n$ □

This allows us to fix $\gamma := (n + 1)^6 4^n$. Now we fix s_0. We prove a lemma showing that for every number γ there is an input from which we can reach a configuration with at least γ agents in each state (we assume wlog that every state can be populated from some input, otherwise we can remove the state). The proof is in the Appendix.

Lemma 9. *For every $\gamma \in \mathbb{N}$ there exists an $s_0 \leq \gamma n 2^n$ such that $IC(s_0) \xrightarrow{*} m$ for some configuration $m \geq \gamma$.*

This allows us to fix $s_0 := \gamma n 2^n \leq (n + 1)^7 2^{3n}$. So together with Lemma 7 we finally get:

Lemma 10 (Pumping Lemma)**.** *Let $\gamma := (n + 1)^6 4^n$ and $s_0 := \gamma n 2^n$. Let m_γ be a configuration satisfying $m_\gamma \geq \gamma$ and $IC(s_0) \xrightarrow{*} m_\gamma$, which exists by Lemma 9. If there exist inputs $a, b \geq 2$ such that*

1. *$m_\gamma + IC(a) \xrightarrow{*} \mu + \mathbb{N}^S$ for some 0-base element (μ, S), and*
2. *$IC(b) \overset{*}{\Longrightarrow} \mathbb{N}^S$,*

then $\mathcal{P}$ rejects $(s_0 + a + \lambda b)$ for every $\lambda \geq 0$.

5.2. QCT-sequences

As we did in Section 4, we define a notion of certificate that an input is rejected, in this case an input larger than s_0. Assume $\mathcal{P}$ computes $x \geq \eta$. By Lemma 9, for every i with

$s_0 + i \leq \eta$ there exist sequences of transitions σ_i and π_i, a 0-base element (μ_i, S_i) and a configuration $m_i \in \mathbb{N}^{S_i}$ such that

$$IC(s_0 + i) \xrightarrow{\sigma_i} C_\gamma + IC(i) \xrightarrow{\pi_i} \mu_i + m_i \tag{1}$$

Further, for every i we can assume that the sequence $\sigma_i \pi_i$ has minimal length and, by Lemma 3, that (μ_i, S_i) has norm at most $2^{2(2n+1)!+1}$. Observe that execution (1) proves $IC(s_0 + i) \xrightarrow{*} SC_0$, and so that $\mathcal{P}$ rejects $s_0 + i$. We define the rejection certificate of i as follows.

Definition 7. *Let $\ell := \eta - 1 - s_0$, and let $i = 1, ..., \ell$. The* (rejection) certificate *of i is the tuple $cert(i) := (\mu_i, S_i, m_i, pv_i)$, where μ_i, S_i, m_i are as in (1), and pv_i is the* Parikh vector *of the sequence $\sigma_i \pi_i$, defined as the mapping $pv_i : T \to \mathbb{N}$ that assigns to every transition the number of times it occurs in $\sigma_i \pi_i$. Further, we say that S_i is the* colour *of i. The* certificate sequence *of $\mathcal{P}$ is the sequence $cert = cert(1)cert(2)...cert(\ell)$.*

Observe that the type of a certificate (μ_i, S_i, m_i, pv_i) is $\mathbb{N}^Q \times 2^Q \times \mathbb{N}^Q \times \mathbb{N}^T$, with the constraint that for every $q \in S_i$ we have $\mu_i(q) = 0$, and for $q \notin S_i$ we have $m_i(q) = 0$.

In Section 4 we saw that the certificates introduced there were good linearly controlled sequences, which allowed us to use existing results. For leaderless protocols we proceed in the same way, with the difference that now we cannot resort to the literature, but have to develop the theory ourselves.

Controlled QCT-sequences.

We first show that $Cert$ is also a controlled sequence in a certain sense. The proof is straightforward and can be found in the Appendix.

Lemma 11. *Let $Cert = cert(1)...cert(\ell)$ be the certificate sequence of $\mathcal{P}$, where $cert(i) = (\mu_i, S_i, m_i, pv_i)$. We have $\|\mu_i\|_1 \leq n2^{2(2n+1)!+1}$, $\|\mu_i\|_1 + \|m_i\|_1 = s_0 + i$, and $\|\mu_i\|_1 + \|m_i\|_1 + \|pv_i\|_1 \leq (s_0 + i)^n$ for all $i = 1, ..., \ell$.*

Lemma 11 motivates the next definition:

Definition 8. *Let Q, C, T be disjoint finite sets of* states, colours, *and* transitions. *A QCT*-tuple *is a fourtuple $qct = (\mu, c, m, pv)$, where $\mu, m \in \mathbb{N}^Q$, $c \in C$, and $pv \in \mathbb{N}^T$. A QCT*-sequence *is a finite sequence of certificates. A QCT-sequence $\tau = qct_1, ..., qct_\ell$, where $qct_i = (\mu_i, c_i, m_i, pv_i)$ is* controlled *if there are constants s_0, α, β such that $\|\mu_i\|_\infty \leq \beta$, $\|\mu_i\|_1 + \|m_i\|_1 = s_0 + i$, and $\|\mu_i\|_1 + \|m_i\|_1 + \|pv_i\|_1 \leq (s_0 + i)^\alpha$ for all $0 \leq i \leq \ell$. We call s_0, α, β the* control parameters *of τ and write $I(c) := \{i : c_i = c\}$ for the indices of elements with colour $c \in C$.*

We can now reformulate Lemma 11 as:

Corollary 12. *$Cert$ is a controlled QCT-sequence with $C = 2^Q$, and control parameters $s_0 \leq (n+1)^7 2^{3n}$, $\alpha = n$ and $\beta = n2^{2(2n+1)!+1}$.*

Linear combinations and good controlled QCT-sequences.

We now show that *Cert* satisfies a property playing the same role as "goodness" of linearly controlled sequences, but stronger. Intuitively, this makes it much harder to produce long good sequences, which leads to a triple exponential bound instead of a non-elementary one.

Definition 9. *Let $qct_1, \ldots, qct_s$ be certificates of the same colour c, where $qct_i = (\mu_i, c, m_i, pv_i)$. A tuple $(\mu, m, pv) \in \mathbb{R}^Q \times 2^Q \times \mathbb{R}^Q \times \mathbb{R}^T$ is a* linear combination *of $qct_1, \ldots, qct_s$ if there are coefficients $\lambda_1, \ldots, \lambda_s \in \mathbb{R}$ such that we have $(\mu, m, pv) = \sum_{i=1}^{s} \lambda_i(\mu_i, m_i pv_i)$.*

Let $\tau = (qct_i)_{i=1,\ldots,\ell}$ be a QCT-sequence. A colour $c \in C$ is bad *if there is a linear combination $qct = (\mu, c, m, pv)$ of $(qct_i)_{i \in I(c)}$ such that $\mu = 0$, $m \gneq 0$, and $pv \geq 0$. A QCT-sequence is* bad *if at least one colour is bad, and* good *otherwise.*

Before showing that *Cert* is a good QCT-sequence, let us give some intuition for this definition. First of all, let us compare the bad sequences of Section 4 with the ones of Definition 9. In Section 4, certificates were triples (μ_i, c_i, m_i), while now they have an extra component (μ_i, c_i, m_i, pv_i). To ease the comparison, ignore the pv component for the moment. A sequence of certificates is bad in the sense of Section 4 if there are indices $i < j$ such that $c_i = c_j$ (i.e. the certificates have the same colour), $\mu_i = \mu_j$, and $m_j \geq m_i$. So we have $\mu_j - \mu_i = 0$ and $m_j - m_i \geq 0$, which implies $m_j - m_i \gneq 0$ (if $m_j - m_i = 0$ then $\mu_i + m_i = \mu_j + m_j$, which can only occur if $i = j$). It follows that the linear combination $(\mu, m, pv) := -(\mu_i, m_i, pv_i) + (\mu_j, m_j, pv_j)$ satisfies the conditions of Definition 9, and so the sequence is also bad in the sense of this definition. But Definition 9 is far more permissible. The sequence is still bad if, for example, we find indices i_1, i_2, j_1, j_2 whose certificates have the same colour and μ-component, and satisfy $m_{j_1} + m_{j_2} \geq m_{i_1} + m_{i_2}$; more generally, it is even enough to find (distinct) multisets of indices I and J satisfying $|I| = |J|$ and $\sum_{i \in J} m_j \geq \sum_{i \in I} m_i$. So, loosely speaking, while in Section 4 we must wait until we see $m_i \leq m_j$ for some indices $i < j$ to declare badness, now it suffices to find two multisets I and J of the same size satisfying $\sum_{i \in I} m_i \leq \sum_{j \in J} m_j$. Intuitively, this makes it much harder to construct a long good sequence, leading to a triple exponential bound on the maximal length of good sequences, instead of the non-elementary bound of Section 4.

Lemma 13. *Cert is a good QCT-sequence.*

Proof. Assume *Cert* is bad. Then there is a bad colour c and a linear combination (μ, m, pv) of $\{cert(i) : i \in I(c)\}$ that satisfies the conditions of Definition 9. We prove that there exist inputs a and b fulfilling the conditions of the Pumping Lemma (Lemma 10), which contradicts the assumption that $\mathcal{P}$ computes $x \geq \eta$.

Let $(\mu_i, c, m_i, pv_i) := cert(i)$ for $i \in I(c)$ and let $y : I(c) \to \mathbb{R}$ denote the coefficients of the linear combination (μ, m, pv), meaning that we have $\sum_i y_i \mu_i = 0$ and $\sum_i y_i m_i \gneq 0$ and $\sum_i y_i pv_i \geq 0$. These conditions are invariant under scaling of y, so we may assume wlog that $y_i \in \mathbb{Z}$ for $i \in I(c)$.

As we already noted, potential reachability depends only on the Parikh vector of the transition sequence. So we will extend $\Rightarrow$ to Parikh vectors by writing $m \overset{pv}{\Longrightarrow} v$ for $pv \in \mathbb{N}^T$ if $m \overset{\sigma}{\Rightarrow} v$ for some sequence $\sigma \in T^*$ with $\overrightarrow{\sigma} = pv$. Note that $m \overset{pv}{\Longrightarrow} v$ is thus equivalent to $m + \sum_{t \in T} pv_t \Delta(t) = v$.

Recall that due to Definition 7, we have

$$IC(s_0 + i) \xrightarrow{\sigma_i} C_\gamma + IC(i) \xrightarrow{\pi_i} \mu_i + m_i \tag{$*$}$$

for every $i \in I(c)$ and sequences $\sigma_i, \gamma_i \in T^*$, where $pv_i = \overrightarrow{\sigma_i} + \overrightarrow{\pi_i}$.

Let us now define inputs a and b fulfilling the conditions of the Pumping Lemma (Lemma 10). For a we simply pick any element $j \in I(c)$ and set $a := j$. By ($*$), condition (1) of Lemma 10 holds for a and $(\mu, S) := (\mu_j, c)$. It remains to prove (2). Set $b := \sum_i y_i(s_0 + i)$ and $pv := \sum_i y_i pv_i$. Since $\sum_i y_i \mu_i = 0$ and $\sum_i y_i m_i \gneq 0$ and $\sum_i y_i pv_i \geq 0$, we have

$$IC(b) = IC\Big(\sum_i y_i(s_0 + i)\Big) = \sum_i y_i IC(s_0 + i) \overset{pv}{\Longrightarrow} \sum_i y_i(\mu_i + m_i) = \sum_i y_i m_i \gneq 0$$

As $m_i \in \mathbb{N}^c$ for $i \in I(c)$ we get $IC(b) \overset{*}{\Rightarrow} \mathbb{N}^c \setminus \{0\}$. Transitions preserve the total number of agents, so $b > 0$. $\square$

5.3. A Bound on the Length of Good QCT-sequences

We obtain a bound on the length of a good controlled QCT-sequence with control parameters s_0, α, β. More precisely, our goal is to prove the following theorem:

Theorem 14. *The length ℓ of a good QCT-sequence with control parameters s_0, α, and β satisfies*

$$\log \ell \leq (\log \beta + 1 + \alpha \log(s_0 + 1))(3 + \alpha)^{|C|(2|Q|+|T|)}$$

Observe that this is is purely combinatorial question, motivated by, but independent from, population protocols.

Notation. *We collect a number of notations used in the rest of the section.*

- *τ denotes a QCT-sequence with control parameters s_0, α, β.*
- *$c \in C$ denotes an arbitrary colour of τ.*
- *$I(c)$ denotes the set of indices of the elements of τ of colour c.*
- *for $i \in I(c)$, $qct_i = (\mu_i, c, m_i, pv_i)$ denotes the i-th element of τ.*
- *for $i \in I(c)$, u_i denotes the concatenation of the vectors m_i and pv_i, for which we use the notation $u_i = \binom{m_i}{pv_i}$.*

- *$I^*(c) \subseteq I(c)$ denotes the the set of indices $i \in I(c)$ s.t. $\binom{u_i}{\mu_i}$ is linearly independent from $\{\binom{u_j}{\mu_j} : j \in I(c), j < i\}$.*

We proceed in several steps:

- In Section 5.3.1 we use Farkas's Lemma to construct a certificate of goodness for a colour c. A certificate of goodness is a mapping that assigns a real number, called a *weight*, to each dimension of μ_i and u_i. The mapping itself is called a *weighting*. We show how to compute *basic weightings* as the unique solution of a system of equations (Lemma 16).
- In Section 5.3.2 we bound the size of a basic weighting, and transform this bound into a bound on the length of τ (Lemma 19). However, the bound still depends on the size of the vectors u_i, with $i \in I^*(c)$.
- In Section 5.3.3 we remove this dependence and prove Theorem 14.

5.3.1. Certifying Goodness with Weightings

We start by formally defining weightings.

Definition 10. *A vector (y, z), where $y \in \mathbb{R}^Q$ and $z \in \mathbb{R}^Q \times \mathbb{R}^T$ is a weighting for the colour c, also called a c-*weighting*, if $z \geq 0$ and $y^\top \mu_i + z^\top u_i = -(s_0 + i)$ for all $i \in I(c)$.*

We use Farkas' Lemma to prove that the existence of a c-weighting is a certificate of goodness for colour c.

Lemma 15. *A colour c is good iff it has a weighting.*

Proof. As stated in Definition 9, c is a bad colour iff

$$\exists x \in \mathbb{R}^{I(c)} : \sum_i x_i \mu_i = 0 \text{ and } \sum_i x_i t_i \geq 0 \text{ and } \sum_i x_i m_i \gneq 0 \tag{1}$$

Let $A_1 := ((\mu_i^\top)_{i \in I(c)})^\top$ be the matrix where column i is μ_i and $A_2 := ((u_i^\top)_{i \in I(c)})^\top$. Now (1) is equivalent to

$$\exists x \in \mathbb{R}^{I(c)} : A_1 x = 0 \text{ and } A_2 x \geq 0 \text{ and } \Big\|\sum_i x_i m_i\Big\|_1 > 0 \tag{2}$$

Recall that by the definition of a QCT-sequence we have $\mathbf{1}^\top(\mu_i + m_i) = \|\mu_i\|_1 + \|m_i\|_1 = s_0 + i$. If we assume further that $A_1 x = 0$ and $A_2 x \geq 0$ hold, we can simplify $\|\sum_i x_i m_i\|_1 > 0$ as follows.

$$\begin{aligned} 0 &< \Big\|\sum_i x_i m_i\Big\|_1 = \mathbf{1}^\top \sum_i x_i m_i = \mathbf{1}^\top \Big(\sum_i x_i m_i + \sum_i x_i \mu_i\Big) \\ &= \sum_i x_i \mathbf{1}^\top (\mu_i + m_i) = \sum_i x_i (s_0 + i) =: b^\top x \end{aligned}$$

where in the last step we define $b \in \mathbb{R}^{I(c)}$ as $b_i := s_0 + i$ for $i \in I(c)$. Hence we now have

$$\exists\, x \in \mathbb{R}^{I(c)} : A_1 x = 0 \text{ and } A_2 x \geq 0 \text{ and } b^\top x > 0 \tag{3}$$

Applying a variant of Farkas' Lemma we can transform (3) into the system (4) that has a solution if, and only if, (3) does not:

$$\exists\, y \in \mathbb{R}^{Q}, z \in \mathbb{R}^{Q \cup T} : A_1^\top y + A_2^\top z = -b \text{ and } z \geq 0 \tag{4}$$

The sequence τ is bad iff (1) is feasible. Moreover, (4) is equivalent to (y, z) being a c-weighting. □

A good colour may have multiple weightings, even an infinite convex set of weightings. Similarly to basic solutions of a linear program, we introduce *basic weightings* of a colour, whose size we will bound using simple linear algebra. Recall that $I^*(c)$ denotes the the set of indices $i \in I(c)$ s.t. $\binom{u_i}{\mu_i}$ is linearly independent from $\{\binom{u_j}{\mu_j} : j \in I(c), j < i\}$. Two properties of a basic weighting are of interest: (1) it is the unique solution of a linear system of equations, and (2) it has at most $|I^*(c)|$ nonzero components. The proof is a straightforward application of well-known properties of linear inequalities, and is given in the appendix.

Lemma 16. *Let c be a good colour. Then there are $Y \subseteq Q, Z \subseteq Q \cup T$ with $|Y| + |Z| = |I^*(c)|$ such that the system $y^\top \mu_i + z^\top u_i = -s_0 - i$, for all $i \in I^*(c)$, has a unique solution $y \in \mathbb{R}^Y, z \in \mathbb{R}^Z$, and (y, z) is a c-weighting. We refer to such a (y, z) as* basic *c-weighting.*

5.3.2. A first bound

The next step is showing that the existence of a basic weighting implies an upper bound on the length of the QCT-sequence. We begin by showing a general bound on a unique solution to a linear system of equations. Again, the proof is routine linear algebra, and can be found in the Appendix.

Lemma 17. *Let $Ax = b$ denote a linear system of equations with unique solution x, where $A \in \mathbb{Z}^{d \times d}$, and let $g(i) \geq \log \max\{|A_{ij}| : j\} \cup \{|b_i|\}$ denote an upper bound of each row i. Then $\log\|x\|_\infty \leq W(g, d)$, where*

$$W(g, d) := 2^{d-1} - 1 + \sum_{t=1}^{d-1} 2^{d-1-t} g(t) + g(d)$$

We now use the previous lemma to prove an upper bound on the components of some c-weighting, for each colour c, based on the sizes of the linearly independent vectors μ_i, u_i with $i \in I^*(c)$. To refer to these sizes, we set $\{l_1, ..., l_d\} := I^*(c)$ with $l_1 < ... < l_d$, and define $g_c(i) := \log(\|\mu_{l_i}\|_1 + \|u_{l_i}\|_1)$ for $i = 1, ..., d$. We remark that Definition 9 immediately gives the estimate $g_c(i) \leq \alpha \log(l_i)$.

Lemma 18. *For each colour c and $d := |I^*(c)|$, there is a c-weighting (y, z) with $\log\|\binom{y}{z}\|_\infty \le W(g_c, d)$.*

Proof. Lemma 16 allows us to construct a c-weighting as the solution to a specific set of linear equations. In particular, we set A to the matrix with $A_{ij} := (\mu_i)_j$ for $i \in I^*(c), j \in Y$ and $A_{ij} := (u_i)_j$ for $i \in I^*(c), j \in Z$, and define b as $b_i = -s_0 - i$ for $i \in I^*(c)$. Then $A\binom{y}{z} = b$ has as unique solution $y \in \mathbb{R}^Y, z \in \mathbb{R}^Z$ where (y, z) is a c-weighting. Now our desired bound follows simply by applying Lemma 17. (Note that $|b_i| = s_0 + i \le (s_0 + i)^\alpha$ holds.) □

From this upper bound we can derive a bound on the length of the sequence (restricted to a specific colour c), using that the weights for the u_i must be nonnegative.

Lemma 19. *For any colour c and $d := |I^*(c)|$, we have $\log \max I(c) \le \log \beta + W(g_c, d)$.*

Proof. Let (y, z) denote a c-weighting fulfilling the bound of Lemma 18. Hence for every $i \in I(c)$ we have $y^\top \mu_i + z^\top u_i = -s_0 - i$. We know that $z^\top u_i \ge 0$ as $z, u_i \ge 0$, so $i \le -y^\top \mu_i - s_0 \le \|y\|_\infty \|\mu_i\|_1$. By Definition 9, $\|\mu_i\|_1 \le \beta$, which we can plug into the bound of Lemma 18 to get the desired statement. □

5.3.3. The final bound

The bound of Lemma 19 still depends on g_c, i.e. the sizes of the elements with indices in $I^*(c)$. We now show how to move from this bound to the one of Theorem 14. The proof that the expression of Theorem 14 is indeed a bound proceeds by induction on d, i.e. assuming that the bound is correct when $I^*(c)$ contains d linearly independent vectors, we show that it remains correct when it contains $d + 1$. For this, observe that in controlled sequences a bound on the length of the sequence yields a bound on the size of its vectors. So we use the sizes of the first d linearly independent vectors to derive a bound on the length of the sequence until the $(d + 1)$-th dimensional vector, which yields a bound on the size of this vector.

There is a slight complication in that the induction needs to be performed for all colours at one, instead of separately for each colour. Our induction variable is thus the total number of linearly independent vectors (of all colours) which we refer to as P. The induction hypothesis also needs to be chosen carefully. We use that the upper bound on $\max I(c)$ (from Lemma 19) is bounded by $f(P)$ for a suitable function f.

Theorem 14. *The length ℓ of a good QCT-sequence with control parameters s_0, α, and β satisfies*

$$\log \ell \le (\log \beta + 1 + \alpha \log(s_0 + 1))(3 + \alpha)^{|C|(2|Q|+|T|)}$$

Proof. Let $d_c := |I^*(c)|$ for $c \in C$ and $P := \sum_c d_c \le |C|(2|Q| + |T|)$. We will prove the stronger statement that $G_c(d_c) \le f(P)$ for *all* colours c with $d_c > 0$, where

$$f(P) := (\log \beta + 1 + \alpha \log(s_0 + 1))(3 + \alpha)^{P-1}$$

$$G_c(r) := \log \beta + 2^{r-1} + \sum_{t=1}^{r-1} 2^{r-1-t} g_c(t) + g_c(r)$$

This is a stronger statement due to Lemma 19 showing that $\log l \leq G_c(d_c)$ for *some* colour c with $I(c) \neq \emptyset$ and thus $d_c > 0$. The proof will proceed by induction on P. In the base case we have $P = 1$ and thus $g_c(1) \leq \alpha \log(s_0 + 1)$ for each $c \in C$, hence $G_c(d_c) \leq f(1)$ (with $d_c \leq 1$).

For the induction step, let $j := \max \bigcup_c I^*(c)$ denote the last index of any linearly independent u_i, i.e. the last index at which P increases; and let c denote the colour of j. For all colours $c' \neq c$, the value of $G_{c'}(d_{c'})$ does not change, so the induction hypothesis yields $G_{c'}(d_{c'}) \leq f(P - 1)$ and thus $G_{c'}(d_{c'}) \leq f(P)$.

For colour c, we use the induction hypothesis to get $\log(j - 1) \leq f(P - 1)$ and $G_c(d_c - 1) \leq f(P - 1)$. By Definition 9 the size of $g_c(d_c)$ (i.e. the vector at index j) can, using the former, be bounded as $g_c(d_c) \leq \alpha \log(s_0 + j) \leq \alpha(f(P - 1) + 1)$. (Here we used $\log(s_0 + 1) \leq f(P - 1)$ and $\log(a + b) \leq 1 + \log a$ for $a \geq b$.) This is then combined with the latter:

$$\begin{aligned} G_c(d_c) &\leq 2G_c(d_c - 1) + g_c(j) \leq 2f(P - 1) + \alpha(f(P - 1) + 1) \\ &\leq (3 + \alpha)f(P - 1) = f(P) \end{aligned}$$

□

5.4. Putting Everything Together

Let us put all the pieces together. Let $\mathcal{P}$ be a leaderless protocol with n states computing a predicate $x \geq \eta$, and let $s_0 := (n + 1)^7 2^{3n}$ be the constant of the Pumping Lemma (Lemma 10). We prove $\eta \leq 2^{2^{2^{\mathcal{O}(n)}}}$.

If $\eta \leq s_0$ then we are done. So assume that $\eta > s_0$.

- Since $\mathcal{P}$ rejects inputs $s_0, s_0 + 1, ..., \eta - 1$, the certificate sequence *Cert* of Definition 7 has length $\ell = \eta - 1 - s_0$.
- By Corollary 12, *Cert* is a controlled *QCT*-sequence with set $C := 2^Q$ of colours, and control parameters s_0, $\alpha := n$, and $\beta = n2^{2(2n+1)!+1}$. Further, by Lemma 13 *Cert* is good.
- By Theorem 13, the length ℓ of *Cert* satisfies

  $$\log \ell \leq (\log \beta + 1 + \alpha \log(s_0 + 1))(3 + \alpha)^{|C|(2|Q|+|T|)}$$

  where $|C| = |2^Q| = 2^n$, $|Q| = n$, and $|T| \leq n^4$ (each transition is determined by a fourtuple of states). This expression is $2^{2^{O(n)}}$.
- So $\eta = \ell + s_0 + 1$ is bounded by $2^{2^{2^{O(n)}}}$.

This yields a triple exponential bound on the busy beaver function for leaderless protocols (see Lemma 23 for the precise bound):

Theorem 20. *$BB(n) \leq 2^{2^{2^{n+5\log n+2}}}$, and so $STATE(n) \in \mathcal{O}(\log \log \log n)$.*

6. Conclusion

We have obtained the first non-trivial lower bounds on the state complexity of population protocols, a fundamental but very hard question about the model. The obvious open questions are to close the gap between the $\Omega(\log\log\log n)$ lower bound and the $\mathcal{O}(\log n)$ upper bound for the leaderless case, and the even larger gap between $\Omega(\log\log n)$ and (roughly speaking), the $\mathcal{O}(\alpha(n))$ upper bound for protocols with leaders, where $\alpha(n)$ is the inverse of the Ackermann function.

References

[1] Sergio Abriola, Santiago Figueira, and Gabriel Senno. Linearizing well quasi-orders and bounding the length of bad sequences. *Theor. Comput. Sci.*, 603:3–22, 2015.

[2] Dan Alistarh, James Aspnes, David Eisenstat, Rati Gelashvili, and Ronald L. Rivest. Time-space trade-offs in population protocols. In *Proc.* 28^{th} *Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2560–2579. SIAM, 2017. doi:10.1137/1.9781611974782.169.

[3] Dan Alistarh, James Aspnes, and Rati Gelashvili. Space-optimal majority in population protocols. In *SODA*, pages 2221–2239. SIAM, 2018.

[4] Dan Alistarh and Rati Gelashvili. Recent algorithmic advances in population protocols. *SIGACT News*, 49(3):63–73, 2018.

[5] Dana Angluin, James Aspnes, Zoë Diamadi, Michael J. Fischer, and René Peralta. Computation in networks of passively mobile finite-state sensors. In *PODC*, pages 290–299. ACM, 2004.

[6] Dana Angluin, James Aspnes, Zoë Diamadi, Michael J. Fischer, and René Peralta. Computation in networks of passively mobile finite-state sensors. *Distributed Computing*, 18(4):235–253, 2006.

[7] Dana Angluin, James Aspnes, Zoë Diamadi, Michael J. Fischer, and René Peralta. Computation in networks of passively mobile finite-state sensors. *Distributed Comput.*, 18(4):235–253, 2006.

[8] Dana Angluin, James Aspnes, and David Eisenstat. Fast computation by population protocols with a leader. *Distributed Comput.*, 21(3):183–199, 2008.

[9] Dana Angluin, James Aspnes, David Eisenstat, and Eric Ruppert. The computational power of population protocols. *Distributed Comput.*, 20(4):279–304, 2007.

[10] A. R. Balasubramanian. Complexity of controlled bad sequences over finite sets of nd. In *LICS*, pages 130–140. ACM, 2020.

[11] A. R. Balasubramanian, Javier Esparza, and Mikhail A. Raskin. Finding cut-offs in leaderless rendez-vous protocols is easy. *CoRR*, abs/2010.09471, 2020. To appear in Proceedings of FOSSACS 2021.

[12] Michael Blondin, Javier Esparza, Blaise Genest, Martin Helfrich, and Stefan Jaax. Succinct population protocols for presburger arithmetic. In *STACS*, volume 154 of *LIPIcs*, pages 40:1–40:15. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.

[13] Michael Blondin, Javier Esparza, and Stefan Jaax. Large flocks of small birds: on the minimal size of population protocols. In *STACS*, volume 96 of *LIPIcs*, pages 16:1–16:14. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018.

[14] Wojciech Czerwinski, Slawomir Lasota, Ranko Lazic, Jérôme Leroux, and Filip Mazowiecki. The reachability problem for petri nets is not elementary. In *STOC*, pages 24–33. ACM, 2019.

[15] Robert Elsässer and Tomasz Radzik. Recent results in population protocols for exact majority and leader election. *Bull. EATCS*, 126, 2018.

[16] Javier Esparza. Petri nets lecture notes, 2019. URL: https://archive.model.in.tum.de/um/courses/petri/SS2019/PNSkript.pdf.

[17] Diego Figueira, Santiago Figueira, Sylvain Schmitz, and Philippe Schnoebelen. Ackermannian and primitive-recursive bounds with dickson's lemma. In *LICS*, pages 269–278. IEEE Computer Society, 2011.

[18] Leszek Gąsieniec and Grzegorz Stachowiak. Enhanced phase clocks, population protocols, and fast space optimal leader election. *J. ACM*, 68(1), 2020.

[19] Christoph Haase. A survival guide to presburger arithmetic. *ACM SIGLOG News*, 5(3):67–82, 2018.

[20] Florian Horn and Arnaud Sangnier. Deciding the existence of cut-off in parameterized rendez-vous networks. In *CONCUR*, volume 171 of *LIPIcs*, pages 46:1–46:16. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2020.

[21] Jérôme Leroux and Sylvain Schmitz. Reachability in vector addition systems is primitive-recursive in fixed dimension. In *LICS*, pages 1–13. IEEE, 2019.

[22] Ken McAloon. Petri nets and large finite sets. *Theor. Comput. Sci.*, 32:173–183, 1984.

[23] Tadao Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580, 1989.

[24] Charles Rackoff. The covering and boundedness problems for vector addition systems. *Theor. Comput. Sci.*, 6:223–231, 1978.

[25] Sylvain Schmitz. Complexity hierarchies beyond elementary. *ACM Trans. Comput. Theory*, 8(1):3:1–3:36, 2016.

[26] Alexander Schrijver. *Theory of Linear and Integer Programming*. John Wiley & Sons, Inc., USA, 1986.

[27] Joachim von zur Gathen and Malte Sieveking. A bound on solutions of linear integer equalities and inequalities. *Proceedings of the American Mathematical Society*, 42(1):155–158, 1978.

A. Appendix

A.1. Proof of Lemma 9

Lemma 9. *For every $\gamma \in \mathbb{N}$ there exists an $s_0 \leq \gamma n 2^n$ such that $IC(s_0) \xrightarrow{*} m$ for some configuration $m \geq \gamma$.*

Proof. Let $Q_i := \{q \in Q : IC(2^i) \to q + m, m \in \mathbb{N}^Q\} \setminus Q_{i-1}$ for $i = 1, 2, \ldots$ and $Q_0 := \{x\}$ the set containing just the initial state. Intuitively, Q_i contains the states reachable starting with 2^i agents, but not 2^{i-1}. We know that each state is reachable from a configuration $IC(s)$ for some $s \in \mathbb{N}$, so $Q = \bigcup_{i \geq 0} Q_i$.

It suffices to prove that $Q_i = \emptyset$ implies $Q_{i+1} = \emptyset$, as then $Q = \bigcup_{i=0}^{n-1} Q_i$ and each $q \in Q$ is reachable starting from $IC(2^n)$. Assume that this is not the case, i.e. there exists some i and $q \in Q_i$ with $Q_{i-1} = \emptyset$ and $IC(2^i) \xrightarrow{\sigma} q + m$ for $m \in \mathbb{N}^Q$. We pick such a q which minimises the length of σ.

This means that the last transition of σ is $(q_1, q_2) \mapsto (q, q_3)$ for some $q_1, q_2, q_3 \in Q$. Additionally, $q_1, q_2 \notin Q_i$ as they are reachable by a shorter sequence. But this implies $q_1, q_2 \in \bigcup_{i=0}^{i-2} Q_i$, i.e. that q_1 and q_2 are each reachable from $IC(2^{i-2})$. Hence $IC(2^{i-1}) \xrightarrow{*} q_1 + q_2 + m' \xrightarrow{*} q + q_3 + m'$ for some $m' \in \mathbb{N}^Q$, contradicting $q \notin Q_{i-1}$. □

A.2. Proof of Lemma 11

Lemma 11. *Let $Cert = cert(1) \ldots cert(\ell)$ be the certificate sequence of $\mathcal{P}$, where $cert(i) = (\mu_i, S_i, m_i, pv_i)$. We have $\|\mu_i\|_1 \leq n 2^{2(2n+1)!+1}$, $\|\mu_i\|_1 + \|m_i\|_1 = s_0 + i$, and $\|\mu_i\|_1 + \|m_i\|_1 + \|pv_i\|_1 \leq (s_0 + i)^n$ for all $i = 1, \ldots, \ell$.*

Proof. Let $r := s_0 + i$. Since $IC(s_0 + i) \xrightarrow{*} \mu_i + m_i$ we have $\|\mu_i\|_1 + \|m_i\|_1 = s_0 + i = r$. Further, $\|\mu_i\|_1 \leq n\|\mu_i\|_\infty \leq \beta$ follows from Lemma 3. For $\|pv_i\|_1$ we know that it is the number of transitions of a shortest execution leading from $IC(r)$ to $\mu_i + m_i$. Since a shortest execution visits a configuration at most once, the length is bounded by the number of configurations with r agents, which is equal to $\binom{n+r-1}{r}$. Using $r \geq 3, n \geq 2$ we obtain:

$$\begin{aligned}\|\mu_i\|_1 + \|m_i\|_1 + \|pv_i\|_1 &\leq r + \tbinom{n+r-1}{r} = r + \prod_{j=1}^{n-1} \frac{r+j}{j} \\ &\leq r + (r+1)r^{n-2} \leq r^{n-2}(2r+1) \leq r^n\end{aligned}$$

□

A.3. Proof of Lemma 16

We need a well-known elementary result from linear algebra

Theorem 21 ([26, Theorem 7.1])**.** *Let $m, n \in \mathbb{N}$, and let $a_1, ..., a_m, b \in \mathbb{R}^n$ denote vectors. We write $t := \operatorname{rank}\{a_1, ..., a_m, b\}$ for the dimension of the subspace spanned by $a_1, ..., a_m$ and b. Then exactly one of the following holds.*

1. *b is a nonnegative linear combination of linearly independent $a_1, ..., a_m$.*
2. *There is a $c \in \mathbb{R}^m$ with $c^\top b < 0$ and $c^\top a_i \geq 0$ for $i = 1, ..., m$, where $c^\top a_i = 0$ for $t-1$ linearly independent a_i.*

The following is an immediate consequence:

Corollary 22. *Let $A \in \mathbb{R}^{n\times m}$ and $P := \{x \in \mathbb{R}^m : Ax = b, x \geq 0\}$ be nonempty. Then it has a solution $x \in P$ with at most n nonzero components.*

Proof. Let a_i denote the i-th column of A. If statement (2) of Theorem 21 were to hold, then for any $x \in P$ we would have $0 > c^\top b = c^\top Ax$ where both $c^\top A$ and x are nonnegative. This is a contradiction, so (1) must hold instead, which directly implies the desired statement, as A has at most n linearly independent columns. □

Now we proceed to prove the Lemma.

Lemma 16. *Let c be a good colour. Then there are $Y \subseteq Q, Z \subseteq Q \cup T$ with $|Y| + |Z| = |I^*(c)|$ such that the system $y^\top \mu_i + z^\top u_i = -s_0 - i$, for all $i \in I^*(c)$, has a unique solution $y \in \mathbb{R}^Y, z \in \mathbb{R}^Z$, and (y, z) is a c-weighting. We refer to such a (y, z) as* basic *c-weighting.*

Proof. By definition, $y \in \mathbb{R}^Q, z \in \mathbb{R}^Q \times \mathbb{R}^T$ is a c-weighting iff $y^\top \mu_i + z^\top u_i = -s_0 - i$ for all $i \in I(c)$ and $z \geq 0$. We know that this system of linear inequalities is feasible as a c-weighting exists, so its set of solutions does not change if we consider only linearly independent rows, and we get $y^\top \mu_i + z^\top u_i = -s_0 - i$ for $i \in I^*(c)$, $z \geq 0$. The statement then follows by considering a basic solution of the corresponding linear program.

For completeness, we provide an alternative argument involving Corollary 22. Let y, z denote a solution to the above system, A the matrix where the i-th row is $[\mu_i \; -\mu_i \; u_i]$ (i.e. the concatenation of $\mu_i, -\mu_i, u_i$) for $i \in I^*(c)$, and set $x := [y_+ \; y_- \; z]$, where $y_+, y_- \geq 0$ are a decomposition of y into positive and negative components fulfilling $y = y_+ - y_-$. Then $Ax = b$, where $b_i := -s_0 - i$. Applying Corollary 22 we then find a solution x with at most $|I^*(c)|$ nonzero components. Picking an x with minimal number of nonzero component, then yields corresponding y, z with a total of at most $|I^*(c)|$ nonzero components. We then define Y and Z as the support of y and z, respectively. If the solution (y, z) is not unique, then it would also be possible to construct a solution (y', z') and thus x' with smaller support, but that would contradict our choice of x. □

A.4. Proof of Lemma 17

Lemma 17. *Let $Ax = b$ denote a linear system of equations with unique solution x, where $A \in \mathbb{Z}^{d \times d}$, and let $g(i) \geq \log \max\{|A_{ij}| : j\} \cup \{|b_i|\}$ denote an upper bound of each row i. Then $\log\|x\|_\infty \leq W(g, d)$, where*

$$W(g, d) := 2^{d-1} - 1 + \sum_{t=1}^{d-1} 2^{d-1-t} g(t) + g(d)$$

Proof. We perform at most $m-1$ iterations of Gaussian elimination to solve for $p := \arg\max_i x_i$, the largest component of x (modifying A in the process). In iteration $i = 1, ..., m-1$, let v denote the i-th row of A. We know that $v = 0$ cannot occur, as A has full rank, so either the only nonzero element of v is p and we solve directly for x_p, or we use v to eliminate one variable from the rest of A, taking care to leave only integer elements. For example, to eliminate element i from another row v', we would update that row to $v_i v' - v'_i v$ (similarly for the right-hand side b). Let

$$a_{ij} := \log \max \{|A_{jt}| : t\} \cup \{|b_j|\}$$

denote the logarithm of the maximum absolute value of the j-th row of the linear system after iteration i, for $j = 1, ..., m$ and $i = 0, ..., m-1$. We then claim

$$a_{ij} \leq g(j) + 2^i - 1 + \sum_{t=1}^{i} 2^{i-t} g(t)$$

For $i = 0$, i.e. before the first iteration, this reduces to $a_{0j} \leq g(j)$, which holds. Using row i to eliminate an element from row j in iteration i means that we get $2^{a_{ij}} \leq 2^{a_{i-1,j}} 2^{a_{i-1,i}} + 2^{a_{i-1,i}} 2^{a_{i-1,j}}$, and therefore

$$\begin{aligned}
a_{ij} &\leq 1 + a_{i-1,j} + a_{i-1,i} \\
&\leq g(j) + 1 + 2 \cdot (2^{i-1} - 1) + \sum_{t=1}^{i-1} 2^{i-1-t} g(t) + \sum_{t=1}^{i-1} 2^{i-1-t} g(t) + g(i) \\
&= g(j) + 2^i - 1 + \sum_{t=1}^{i} 2^{i-t} g(t)
\end{aligned}$$

As all coefficients remain integral during the computation, we finally have $\log\|x\|_\infty \leq a_{k-1,k}$, which is the bound we wanted to show. □

B. Proof of the Final Bound

Lemma 23. *Let $|Q| = n \geq 2$, $|T| \leq n^4$, $s_0 := (n+1)^7 2^{3n}$, $\ell := \eta - 1 - s_0$, $|C| = 2^n$, $\alpha := n$, $\beta := n 2^{2(2n+1)!+1}$, and*

$$\log \ell \leq (\log \beta + 1 + \alpha \log(s_0 + 1))(3 + \alpha)^{|C|(2|Q|+|T|)}$$

Then

$$\log \log \log \eta \leq n + 5 \log(n) + 2$$

Proof. We write $\eta_0 := \eta$ and $\eta_{i+1} := \log \eta_i$, and set $P := |C|(2|Q| + |T|) = 2^n(2n + n^4)$.

$$\begin{aligned}
\eta_1 &= \log(\ell + 1 + s_0) \\
&\leq (\log \beta + 2 + \alpha \log(s_0 + 1))(3 + \alpha)^P \\
&\leq (2(2n + 1)! + \log(n) + 3 + n \log((n + 1)^7 2^{3n} + 1))(3 + n)^P \\
&\leq ((2n + 2)! + 7n \log(n + 1) + 3n)(3 + n)^P \\
&\leq ((2n + 2)! + 4n(2n + 1))(3 + n)^P && (1) \\
&\leq (2n + 4)!(3 + n)^P && (\log(n + 1) \leq n)
\end{aligned}$$

At (1) we use $\log(a + b) \leq \log(a) + 1$ for $0 < b \leq a$.

$$\begin{aligned}
\eta_2 &\leq \log((2n + 4)!) + P \log(3 + n) \\
&\leq (2n + 4) \log(n + 3) + P \log(3 + n) && (n! \leq (\tfrac{n}{2})^n) \\
&= \log(n + 3)(2^n(2n + n^4) + 2n + 4)
\end{aligned}$$

Now we will use (2) $\log(a + b) \leq \log(a) + b/\ln(2)a \leq \log(a) + 3b/2a$ for $a, b > 0$ and (3) $\log\log(n + 3) \leq \log\log(n) + \frac{5}{4}$, where the latter is due to $\log\log 5 \leq \frac{5}{4}$.

$$\begin{aligned}
\eta_3 &\leq \log\log(n + 3) + \log(2^n(2n + n^4) + 2n + 4) \\
&\leq \log\log(n) + \tfrac{5}{4} + \tfrac{12}{40} + n + \log(2n + n^4)) && (2), (3) \\
&\leq \log\log(n) + \tfrac{5}{4} + \tfrac{12}{40} + \tfrac{6}{16} + n + 4\log(n) && (2) \\
&\leq n + 5\log(n) + 2
\end{aligned}$$

□