Control Synthesis for High-Dimensional Systems With Counting Constraints

Petter Nilsson, Student Member, IEEE, and Necmiye Ozay, Member, IEEE.

Abstract—General purpose correct-by-construction synthesis methods are limited to systems with low dimensionality or simple specifications. In this work we consider highly symmetrical counting problems and exploit the symmetry to synthesize provably correct controllers for systems with tens of thousands of states. The key ingredients of the solution are an aggregate abstraction procedure for mildly heterogeneous systems and subsequent formulation of counting constraints as linear inequalities.

Index Terms—Abstractions, correct-by-construction control synthesis, symbolic methods, switched systems.

I. INTRODUCTION

AUTOMATED controller synthesis for systems subject to an a priori given specification is an attractive means of controller design; such correct-by-construction synthesis methods have attracted considerable interest in the past years [1]. However, all methods that are capable of solving general problems face fundamental limitations in terms of the system dimensionality and specification complexity that can be handled. A large body of work resorts to abstraction procedures [2], where a finite representation of the system is obtained for which synthesis techniques for finite-state systems can be applied, but such methods quickly run into the curse of dimensionality since, in general, an exponential number of discrete states is required to represent a high-dimensional system with maintained accuracy. Other methods operate directly on a continuous state space but also “blow up” when the representation and treatment of high-dimensional sets become overwhelmingly expensive.

Ways to overcome these limitations have been proposed. A large problem can sometimes be decomposed into several lower-dimensional problems [3], [4], where each subproblem is easier to solve. Such methods typically require the decomposition to be given, and that the coupling between the subproblems is relatively weak. Other work has exploited monotonicity [5], [6] to alleviate the curse of dimensionality when constructing an abstraction.

In this work we explore a different approach to tackle high-dimensionality: exploitation of symmetries. The system we consider is the aggregate system consisting of $N$ almost-homogeneous switched subsystems, where $N$ can be very large. Due to homogeneity the system exhibits symmetries in the dynamics which can be leveraged to enable synthesis for a special type of symmetric constraints called counting constraints. Like many existing methods in the literature, our solution approach is abstraction-based, but with the crucial difference that we construct only one abstraction that is used for all subsystems, thus roughly reducing the number of states in the abstraction from $O(1/\eta^{Nn_x})$ in the naive approach to $O(1/\eta^{n_x})$, where $n_x$ is the dimension of a single subsystem and $\eta$ the precision. We are able to account for mild heterogeneity among the subsystems by assuming certain stability conditions. To the best of our knowledge, this idea of utilizing a single abstraction for multiple subsystems is novel and allows us to solve synthesis problems of very high dimension that exhibit these symmetries—we demonstrate the method on a 10,000-dimensional switched system with $2^{10,000}$ modes. If there are several distinct classes of subsystems (i.e., strong heterogeneity), one abstraction can be constructed for each class; and complexity increases linearly with the number of classes.

This work is motivated by the problem of scheduling of thermostatically controlled loads (TCLs), that include air conditioners, water heaters, refrigerators, etc., that operate around a temperature set point by switching between being on and off. TCL owners are typically indifferent to small temperature perturbations and accept temperatures in a range around their desired set point; this range is called the dead band. The idea behind TCL scheduling is that an electric utility company can leverage the implied flexibility—which becomes meaningful for large collections of TCLs—to shape aggregate demand on the grid. Previous work has resulted in control algorithms based on broadcasting a universal set-point temperature [7], Markov chain bin models [8], [9], and priority stacks [10], with the objective to track a power signal. However, they do not take into account any hard constraints on aggregate power consumption (i.e., the number of TCLs in mode on), which is crucial to avoid overloading the grid or underutilizing the power generated by renewable resources.

The fundamental problem in TCL scheduling is to simultaneously control local safety constraints (i.e., maintain each TCL in its deadband), and global aggregate constraints. These constraints are special instances of counting constraints, which is the type of constraints we consider in this work. The TCL scheduling example will be used throughout the paper to motivate our discussion:

Example. Let $\{x_n\}_{n \in [N]}$ be the states of a family of $N$ switched systems of the following form:

$$\frac{dx_n(t)}{dt} = f_{\sigma_n(t)}(x_n(t), d_n(t)), \quad \sigma_n(t) \in \{\text{on}, \text{off}\},$$

where $f_{\text{on}}$ and $f_{\text{off}}$ are functions $\mathbb{R} \times D \rightarrow \mathbb{R}$ for some bounded disturbance set $D$. For a desired dead band $[a, b]$,
and bounds $[K, \overline{K}]$ on the aggregate number of TCLs that are in mode on. Synthesize switching protocols that enforce

$$x_n(t) \in [a, b] \text{ for all } n \in [N] \text{ and all } t \in \mathbb{R}^+,$$

$$\text{For all } t \in \mathbb{R}^+, \text{ at least } K \text{ and at most } \overline{K} \text{ of the TCLs are in mode on.}$$

A potential solution to this type of problem is presented in [11], [12], which proposes a linear programming-based schedule search for bounded-rate systems. While that approach allows for more general set constraints and has polynomial complexity in the number of permitted aggregate modes, it still becomes impractical for large-scale systems since the number of aggregate modes grows exponentially. Also related is recent work on intersection clearing controllers [13], where integer programs are used to find a finite-horizon schedule that allows cars to clear an intersection. Our work can be seen as a marriage between the abstraction approach via “bins” previously used in TCL literature, and schedule search via integer programming—generalized to arbitrary incrementally stable systems and an infinite horizon.

A preliminary version of this work appeared in [14] where the mode-counting problem was introduced for homogeneous collections of systems. In this paper we generalize the counting problem to encompass both mode- and state-counting constraints; make the construction of the abstractions robust to model deviations—thus allowing mild heterogeneity; improve the rounding scheme that enables search for solutions via (non-integer) linear programming; and sharpen analytical results. A different extension of counting problems that allows for a richer class of specifications—temporal logic formulas defined over counting propositions—was reported in [15].

### A. Paper Layout

The paper is structured in the following way. Section II introduces notation that is used throughout the paper. The counting problem is introduced in Section III while the subsequent Section IV suggests an abstraction procedure that transforms a continuous-state counting problem into a discrete one, along with results that relate the solvability of the two problem instances. A solution approach to the discrete-state counting problem is presented in Section V in the form of a linear feasibility problem that can either be solved as a counting problem is presented in Section V in the form of a linear feasibility problem that can either be solved as a

**II. Notation and Preliminaries**

In the following paragraphs we introduce some notation that is used throughout the paper. The set of real numbers is denoted $\mathbb{R}$, the set of positive reals $\mathbb{R}_+$, and the set of non-negative integers (i.e., including zero) $\mathbb{N}$. To express a finite set of positive integers, we write $[N] = \{0, \ldots, N - 1\}$. The indicator function of a set $X$ is denoted $\mathbf{1}_X(x)$ and is equal to 1 if $x \in X$ and to 0 otherwise. The identity function on a set $A$ is written as $\mathrm{Id}_A$. For two sets $X$ and $Y$, we write the Minkowski sum as $X + Y = \{x + y : x \in X, y \in Y\}$, and the Minkowski difference $X - Y = \{x : \{x\} \cap Y \subset X\}$. Set complement is denoted $X^C$.

To denote the floor and ceiling of a number we write $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$. We use the same notation for vectors, where the operations are performed component-wise. We write the infinity norm as $\|\cdot\|_\infty$ and the 1-norm as $\|\cdot\|_1$. The $\epsilon$-ball in $p-$norm centered at the point $x$ is denoted as $B_p(x, \epsilon) = \{y : \|y - x\|_p \leq \epsilon\}$. The vector of all 1’s is written as 1. For a function $f : X \rightarrow \mathbb{R}$ with a finite domain we abuse notation and write

$$\|f\|_1 = \sum_{x \in X} |f(x)|$$

for the “1-norm” of the finite image set. The function that is constantly equal to 0 is written 0.

Given an ODE $\frac{d}{dt}x(t) = f(x(t), d(t))$, where $d(t)$ is an uncontrolled input, the corresponding flow operator is denoted $\phi_\mu(t, x, d)$ and has the properties that $\phi_\mu(0, x, d) = x$, $\frac{d}{dt}\phi_\mu(t, x, d) = f(\phi_\mu(t, x, d), d(t))$. So called $\mathcal{KL}$-functions are related to nonlinear stability theory, and are functions $\beta : \mathbb{R}_+ \times \mathbb{R}_+ \rightarrow \mathbb{R}_+$ that are strictly increasing from 0 in the first argument and decreasingly converging to 0 in the second argument.

**A. Transition Systems and Simulations**

We employ the following definition for a transition system, which captures systems with both continuous and discrete state spaces.

**Definition 1.** A transition system (TS) is a tuple $\Sigma = (Q, U, \rightarrow, Y)$, where $Q$ is a set of states, $U$ a set of actions (inputs), $\rightarrow \subset Q \times U \times Q$ a transition relation, and $Y : Q \rightarrow Y$ an output function. We say that $\Sigma$ is a deterministic finite transition system (DFTS) if i) transitions are deterministic, i.e., if $(q, \mu, q') \in \rightarrow$ and $(q, \mu, q'') \in \rightarrow$, then $q' = q''$, and ii) $Q$ is finite.

For simplicity, existence of a transition $(q, \mu, q') \in \rightarrow$ is written $q \xrightarrow{\mu} q'$. By a trajectory of a transition system, we mean a sequence $x(0) x(1) x(2) \ldots$ of states in $Q$ with the property that $x(s) \xrightarrow{\sigma(s)} x(s + 1)$ for some $\sigma(s) \in U$, for all $s \in \mathbb{N}$.

In this paper we restrict attention to finite input sets $U$; such systems are known as switched systems. A switching protocol is a function $\pi$ that generates control inputs from information about the current state. A trajectory generated by a switching protocol $\pi$ is a trajectory where $\sigma(s)$ is generated by $\pi$.

For two systems with the same input space $U$ and normed output space $Y$, we adopt a notion of approximate bisimilarity [2].

1. A switching protocol may also have internal memory states.
Definition 2. Two transition systems \((Q_1, \mathcal{U}_1, \rightarrow_1, Y_1)\) and \((Q_2, \mathcal{U}_2, \rightarrow_2, Y_2)\) are \(\epsilon\)-approximately bisimilar if there exists a relation \(R \subset Q_1 \times Q_2\) such that for all \((q_1, q_2) \in R\),
1) \(\|Y_1(q_1) - Y_2(q_2)\|_\infty \leq \epsilon\),
2) if \(q_1 \xrightarrow{\mu_1} q_1'\), there exists \(q_2 \xrightarrow{\mu_2} q_2'\) s.t. \((q_1', q_2') \in R\),
3) if \(q_2 \xrightarrow{\mu_2} q_2'\), there exists \(q_1 \xrightarrow{\mu_1} q_1'\) s.t. \((q_1', q_2') \in R\).

B. Graphs

The following standard notions are used for a directed graph \(G = (Q, E)\) with node set \(Q\) and edge set \(E \subset Q \times Q\). A path in \(G\) is a list of edges \((q_0, q_1)(q_1, q_2) \ldots (q_{j-1}, q_j)\). The distance between two nodes \(q_0\) and \(q_J\) is the length of the shortest path connecting the two. The graph diameter \(\text{diam}(G)\) is the longest distance between two nodes in the graph. If the first and last nodes in a path are equal, i.e. \(q_J = q_0\), the path is a cycle. A cycle is simple if it visits every node at most one time. For a subset of nodes \(D\) in \(G\) with at most one time. For a subset of nodes \(D\) the path is a cycle counting sets in the form of a product of a set \(X \times U\) restricted to a singleton member of \(U\), which we use interchangeably), it is strongly connected if for each node pair \((q, q')\) in \(D\), there exists a path from \(q\) to \(q'\). Any directed graph can be decomposed into strongly connected components. The period of a subgraph \(D\) is the greatest common divisor of all cycles in \(D\). A subgraph \(D\) is called aperiodic if it has period one.

III. THE COUNTING PROBLEM

We first define the concept of a counting constraint and introduce the counting problem.

Definition 3. A counting constraint \((X, R)\) for a collection of \(N\) subsystems numbered from 0 to \(N - 1\), with the same state space \(Q\) and input space \(U\), is a set \(X = X^Q \times X^U\) with \(X^Q \subset Q\) and \(X^U \subset U\), and a bound \(R\). We say that the counting constraint is satisfied by state-input pairs \((q_n, \mu_n)\), \(n \in [N]\), if the aggregate number of pairs that fall in \(X\) is less than or equal to \(R\):

\[
\sum_{n \in [N]} 1_X(q_n, \mu_n) \leq R.
\]

Remark 1. For simplicity of exposition we restrict attention to counting sets in the form of a product of a set \(Q \times U\) in output space and a set \(X^U\) in mode space. However, our solution method in Section IV can handle more general subsets of \(Q \times U\).

This notion of a counting constraint extends that in [14] by allowing \(X\) to be a subset of \(Q \times U\), instead of being restricted to a singleton member of \(U\). This generalization permits counting constraints that include the state space, as opposed to only counting the number of systems using each action (i.e., mode-counting). A local safety constraint \(x_n(t) \notin S\) for all \(n \in [N]\) can be expressed with the counting constraint \((S \times U, 0)\), therefore there is no need to treat local safety constraints separately as in the cited paper. Furthermore, if a lower bound is desired, a counting constraint for the complement \(X^C\) of \(X\) can be specified:

\[
\sum_{n \in [N]} 1_X(x_n(t), \sigma_n(t)) \geq R
\]

\[
\iff \sum_{n \in [N]} 1_{X^C}(x_n(t), \sigma_n(t)) \leq N - R.
\]

Example (continued). The TCL scheduling constraints [2] and [5] fit into this class of constraints. Let the output space be equal to the state space, i.e. \(Y = \mathbb{R}\) and \(Y(x) = x\). Then the following counting constraints specify that the number of TCLs that are in mode \(\sigma\) at time \(t\) should be in the interval \([K, K]\):

\[
\sum_{n \in [N]} 1_{\mathbb{R}_x(\sigma)}(x_n(t), \sigma_n(t)) \leq K,
\]

\[
\sum_{n \in [N]} 1_{\mathbb{R}_x(\sigma^l)}(x_n(t), \sigma_n(t)) \leq N - K.
\]

In addition, the dead band constraints [2] can be imposed by the counting constraint

\[
\sum_{n \in [N]} 1_{[0, \eta]^c \times \mathbb{R}_x(\sigma^l)}(x_n(t), \sigma_n(t)) \leq 0.
\]

Consider now a family of \(N\) identical subsystems with dynamics given by a transition system and some initial conditions \(x_n(0)\) for \(n \in [N]\). The problem we seek to solve is the following:

Problem 1. Consider \(N\) subsystems, all governed by the same transition system \(\Sigma_{TS} = (Q, \mathcal{U}, \rightarrow, Y)\), where \(\mathcal{U}\) is finite, and assume that initial states \(x_n(0)\) for \(n \in [N]\) and L counting constraints \(\{X_l, R_l\}_{l \in [L]}\) are given. Synthesize individual switching protocols \(\{\pi_n\}_{n \in [N]}\) such that the generated actions \(\sigma_n(0)\) are \(\sigma_n(1)\) \(\sigma_n(2)\) \ldots and trajectories \(x_n(0)x_n(1)x_n(2)\) \ldots for \(n \in [N]\) satisfy the counting constraints

\[
\sum_{n \in [N]} 1_{X_l}(x_n(s), \sigma_n(s)) \leq R_l, \quad \forall s \in \mathbb{N}, \forall l \in [L].
\]

An instance of this problem is referred to as

\[
(N, \Sigma_{TS}, \{x_n(0)\}_{n \in [N]}, \{X_l, R_l\}_{l \in [L]}).
\]

Naïvely, if \(Q\) is an \(n_q\)-dimensional space, the overall system can be viewed as a \(N \times n_q\)-dimensional hybrid system with \(|U|^N\) modes and is thus beyond the reach of common control synthesis techniques that do not exploit the symmetry. It does not matter which subsystems contribute to the summation in (6) for the constraint to be satisfied, so Problem 1 exhibits sub-system permutation symmetry in its specification. In addition, there is also subsystem permutation symmetry in the dynamics since all subsystem states are governed by the same transition system. As an implication there is no need to keep track of identities of individual subsystems—they are all equivalent from both a dynamics- as well as from a specification-point of view. This symmetry (i.e., permutation invariance) is the key to solving large-scale counting problems.

The goal of this work is to synthesize controllers that satisfy counting constraints for a collection of continuous-time switched systems with states \(x_i(t)\), \(i \in [N]\), with dynamics governed by

\[
\frac{dx_i(t)}{dt} = f_{\sigma_i(t)}(x_i(t), d_i(t)), \quad \sigma_i : \mathbb{R} \to \mathcal{U},
\]

for \(x_i(t) \in \mathbb{R}^{n_x}\) and a disturbance signal \(d_i(t)\) assumed to take values in a set \(D\). Although the system description is identical for all \(N\) systems in the family, the disturbance...
signals $d_i$ are not, and can therefore model mild heterogeneity, including parameter variations across system models or modeling inaccuracies. We assume that the vector fields and the disturbance signals satisfy the standard assumptions for existence and uniqueness of solutions.

In addition, we assume that the vector fields exhibit a certain form of stability [16].

**Assumption 1.** For each $\mu \in \mathcal{U}$, the vector field $f_\mu(x, d)$ is $C^1$ in $x$ and continuous in $d$. Furthermore, the nominal system $\frac{d}{dt}x = f_\mu(x, 0)$ is forward complete and incrementally stable. That is, there exists a $K\ell$-function $\beta_\mu$ such that

$$\|\phi_\mu(t, x, 0) - \phi_\mu(t, y, 0)\|_\infty \leq \beta_\mu(\|x - y\|_\infty, t).$$

Note that the vector field $f_\mu(x, d)$ being $C^1$ in $x$ implies that $f_\mu(x, d)$ is Lipschitz in $x$ (with some Lipschitz constant $K_\mu$) when the states are constrained to a compact set.

Furthermore, we assume that the disturbance signal is continuous, and that its effect is bounded in an absolute sense.

**Assumption 2.** The disturbance signal $d : \mathbb{R} \to \mathcal{D}$ is continuous. Furthermore for all $\mu \in \mathcal{U}$, compared to the nominal vector fields without disturbance, the effect of the disturbance is less than $\delta_\mu$:

$$\|f_\mu(x, d) - f_\mu(x, 0)\|_\infty \leq \delta_\mu,$$

for all $d \in \mathcal{D}$.

We have introduced the counting problem only for transition systems, which capture time-sampled continuous-state systems but not systems with continuous time evolution such as (8). The analogue of Problem [1] for a continuous-time system could easily be stated. One key difference is that in a time-sampled system mode switches can only happen at the sampling instants $\tau \mathbb{N}$, where $\tau$ is the sampling time; thus the actuation is constrained. Furthermore, a solution where state counting constraints are violated in between samplings (but satisfied at sample instants) is still a valid solution to the instance (7). If such inter-sample violations are unacceptable, the state-parts of counting sets can be contracted by some margin determined by the dynamics to ensure satisfaction for all $t \in \mathbb{R}_+$ [17].

In the following, we construct the time-sampled analogue of (5) in order to leverage the notion of bisimilarity. We also restrict the domain to a bounded set $X \subset \mathbb{R}^{n_x}$ and let the space-restricted $\tau$-sampled counterpart of (5) confined to $X \subset \mathcal{X}$ be as follows:

$$\Sigma_\tau = (\mathcal{X}, \mathcal{U}, \tau, \text{Id}_{\mathbb{R}^d}),$$

where

$$x \xrightarrow{\mu, \tau} x' \text{ iff } \exists d : [0, \tau] \to \mathcal{D} \text{ s.t. } x' = \phi_\mu(\tau, x, d).$$

The instance

$$(N, \Sigma_\tau, \{x_n(0)\}_{n \in [N]}, \{X_I, R_I\}_{I \in [L_I]})$$

of Problem [1] is referred to as the continuous-state counting problem. Since $\Sigma_\tau$ is a non-deterministic, infinite transition system, this is a difficult problem to solve. In the following we construct a DFTS $\Sigma_{\tau,\eta}$ that is approximately bisimilar to $\Sigma_\tau$, and relate the corresponding problem instances.

**IV. ABSTRACTING THE CONTINUOUS-STATE COUNTING PROBLEM**

We first present an abstraction procedure that creates a finite-state model of (5). Under the assumptions outlined above, we establish approximate bisimilarity between the continuous-state system and its finite-state abstraction, which enables us to state results in Section IV-B relating the solvability of the corresponding counting problems.

**A. Abstraction Procedure**

Consider a system of the form (5). For a state discretization parameter $\eta > 0$ we define an abstraction function $\kappa_\eta : \mathcal{X} \to \mathcal{X}$ as

$$\kappa_\eta(x) = \eta \cdot \left\lfloor \frac{x}{\eta} \right\rfloor + \frac{\eta}{2}.$$

This function is constant on hyperboxes of side $\eta$, and its image of the compact set $\mathcal{X}$ is finite. The abstraction function defines the space-restricted, $(\tau, \eta)$-discretized counterpart of (5) as the transition system

$$\Sigma_{\tau,\eta} = \left(\kappa_\eta(\mathcal{X}), \mathcal{U}, \tau, \eta, \text{Id}_{\mathbb{R}^d}\right),$$

where

$$q \xrightarrow{\mu, \tau, \eta} q' \iff \kappa_\eta(\phi_\mu(\tau, q, 0)) = q'.$$

In essence, the domain $\mathcal{X}$ is partitioned into uniform boxes of size $\eta$ that represent discrete states. As illustrated in Fig. 1, transition relations are established by simulating each mode, without disturbance, during a time $\tau$, starting at the center of the boxes. As opposed to $\Sigma_\tau$, the resulting transition system $\Sigma_{\tau,\eta}$ is finite and deterministic—for each state $q$ and action $\mu$ there exists (at most) one successor state $q'$. This makes the corresponding counting problem easier to solve.

Similarly to a result from [13] we can now show that an abstraction constructed in this way is bisimilar to the time-sampled system (10) if a certain inequality holds.

**Proposition 1.** Assume that Assumptions 1 and 2 hold, i.e., for all $\mu \in \mathcal{U}$ there are Lipschitz constants $K_\mu$, $\ell$-functions $\beta_\mu$, and disturbance effect bounds $\delta_\mu$ associated with the modes of (8). Then, if for all $\mu \in \mathcal{U}$,

$$\beta_\mu(\epsilon, \tau) + \frac{\delta_\mu}{K_\mu}(e^{K_\mu \tau} - 1) + \frac{\eta}{2} \leq \epsilon,$$

the $(\tau, \eta)$-discretized abstraction $\Sigma_{\tau,\eta}$ and the $\tau$-sampled system $\Sigma_\tau$ are $\epsilon$-approximately bisimilar.
Proof. For a grid point \( q \in \kappa_\eta(X) \) and \( x \in X \), consider the relation \( q \sim x \) iff \( \|q - x\|_\infty \leq \epsilon \). This relation evidently satisfies 1) of Definition 2. Item 2) of the same definition is trivially satisfied since all transitions of \( \Sigma_{\tau,\eta} \) are present in \( \Sigma_\tau \) as the special case without disturbance.

What remains is to show that item 3) holds. To this end, assuming that \( q \sim x \), we need to show that \( q' \approx \phi_\mu(\tau, x, d) \) for all permissible \( d : [0, \tau] \to D \), where \( q' \in \kappa_\eta(X) \) is the \( \mu \)-successor of \( q \) in \( \Sigma_{\tau,\eta} \).

From the construction of \( \Sigma_{\tau,\eta} \), \( \|q' - \phi_\mu(\tau, q, 0)\|_\infty \leq \eta/2 \). Furthermore, under the continuity assumptions, it is known that the flow \( \phi_\mu(t, x, 0) \) of \( f_\mu(t, x) \) and the flow \( \phi_\mu(t, x, d) \) for any \( d(t) \) such that \( \max_{t \in [0, \tau]} \|f_\mu(t, x, 0) - f_\mu(t, x, d(t))\|_\infty \leq \delta_\mu \) satisfy \( \|\phi_\mu(t, x, 0) - \phi_\mu(t, x, d)\|_\infty \leq (\delta_\mu / K_\mu)(e^{K_\mu t} - 1) \) [19].

Thus, it follows for any \( d : [0, \tau] \to D \),
\[
\|q' - \phi_\mu(\tau, x, d)\|_\infty \leq \|\phi_\mu(\tau, x, d) - \phi_\mu(\tau, q, 0)\|_\infty + \|\phi_\mu(\tau, q, 0) - q'\|_\infty \\
\leq \delta_\mu / K_\mu (e^{K_\mu \tau} - 1) + \beta_\mu (\|q - x\|_\infty, \tau) + \frac{\eta}{2} \\
\leq \delta_\mu / K_\mu (e^{K_\mu \tau} - 1) + \beta_\mu (\epsilon, \tau) + \frac{\eta}{2} \leq \epsilon.
\]

Hence \( q' \sim \phi_\mu(\tau, x, d) \) which completes the proof. \( \square \)

It is known that the trajectories of \( \epsilon \)-approximately bisimilar systems remain within distance \( \epsilon \) of each other [20]. This fact is the key to the following two results that establish relations between existence of solutions of the counting problem in the continuous-state and finite-state settings.

B. Relations Between the Continuous-State and Discrete-State Counting Problems

If \( \epsilon \)-approximate bisimilarity holds, trajectories of \( \Sigma_{\tau,\eta} \) and \( \Sigma_\tau \) are guaranteed to remain \( \epsilon \)-close when initial states are \( \epsilon \)-close and corresponding actions are chosen. Therefore, solvability of the counting problem is equivalent for the two up to an approximation margin \( \epsilon \). To make precise statements we introduce functions \( G^{+\epsilon} \) that expand (resp. contract) a counting set \( X = X^X \times X^U \) in the state space domain before abstraction:
\[
G^{+\epsilon}(X^X \times X^U) = \kappa_\eta'(X^X \times B_\infty(0, \epsilon)) \times X^U , \\
G^{-\epsilon}(X^X \times X^U) = \kappa_\eta'(X^X \times B_\infty(0, \epsilon)) \times X^U .
\]

Now we can describe how solving the counting problem can be mapped between the time-sampled system \( \Sigma_\tau \) and the time-state abstracted system \( \Sigma_{\tau,\eta} \).

Theorem 1. Let \( \Sigma_\tau \) and \( \Sigma_{\tau,\eta} \) be the time-sampled and time-state discretized systems constructed from a system on the form [9], such that \( \Sigma_\tau \) and \( \Sigma_{\tau,\eta} \) are \( \epsilon \)-approximately bisimilar. Let \( \kappa_\eta \) be the abstraction function for \( \Sigma_{\tau,\eta} \).

If there exists a solution to the instance
\[
(N, \Sigma_\tau, \{\kappa_\eta(x_n(0))\}_{n \in [N]}, \{G^{+\epsilon}(X_I), R_i\}_{i \in [L]}) \tag{13}
\]
of Problem 7 then there exists a solution to the instance
\[
(N, \Sigma_\tau, \{x_n(0)\}_{n \in [N]}, \{X_I, R_i\}_{i \in [L]}) \tag{14}
\]

Proof. Let \( \{\pi_n\}_{n \in [N]} \) be individual switching protocols that solve [13] by generating trajectories \( \xi_n(0)\xi_n(1) \ldots \) and actions \( \sigma_n(0)\sigma_n(1) \ldots \) for \( \Sigma_{\tau,\eta} \). Due to bisimilarity and the \( \eta/2 \)-proximity of initial conditions, the individual trajectories \( \xi_n(0)\xi_n(1) \ldots \) of \( \Sigma_{\tau,\eta} \) and the individual trajectories \( x_n(0)x_n(1) \ldots \) of \( \Sigma_\tau \) satisfy \( \|\xi_n(s) - x_n(s)\|_\infty \leq \epsilon \) for all \( s \in \mathbb{N} \) when the action sequence \( \sigma_n(0)\sigma_n(1) \ldots \) is implemented for both systems. By assumption,
\[
\sum_{n \in [N]} \mathbb{I}_{G^{+\epsilon}(X_I)}(\xi_n(s), \sigma_n(s)) \leq R_I .
\]

Thus for a counting set \( X_I = X^X_I \times X^U_I \),
\[
x_n(s) \in X^X_I \implies \xi_n(s) \in X^X_I \oplus B_\infty(0, \epsilon) \implies \xi_n(s) \in \kappa_\eta(X^X_I \oplus B_\infty(0, \epsilon)) ,
\]

where the last step follows from knowing that \( \xi_n(s) \) only takes values \( x \) such that \( \kappa_\eta(x) = x \). Thus,
\[
\sum_{n \in [N]} \mathbb{I}_{X_I}(x_n(s), \sigma_n(s)) \leq \sum_{n \in [N]} \mathbb{I}_{G^{+\epsilon}(X_I)}(\xi_n(s), \sigma_n(s)) \leq R_I ,
\]

which shows that the corresponding counting constraint in [14] is satisfied. \( \square \)

Theorem 2. Under the same assumptions as in Theorem 7 if there is no solution to the instance
\[
(N, \Sigma_{\tau,\eta}, \{\kappa_\eta(x_n(0))\}_{n \in [N]}, \{G^{-\epsilon(\epsilon + \frac{\eta}{2})}(X_I), R_i\}_{i \in [L]}) \tag{15}
\]
of Problem 7 then there is no solution to the instance
\[
(N, \Sigma_\tau, \{x_n(0)\}_{n \in [N]}, \{X_I, R_i\}_{i \in [L]}) . \tag{16}
\]

Proof. Suppose for contradiction that there is a solution to [16] but not to [15]. Let \( \{\pi_n\}_{n \in [N]} \) be individual switching policies that solve [15] by generating trajectories \( x_n(0)x_n(1) \ldots \) and actions \( \sigma_n(0)\sigma_n(1) \ldots \) for \( \Sigma_\tau \). Due to bisimilarity and the \( \eta/2 \)-proximity of initial conditions, the individual trajectories \( \xi_n(0)\xi_n(1) \ldots \) of \( \Sigma_{\tau,\eta} \) and the individual trajectories \( x_n(0)x_n(1) \ldots \) of \( \Sigma_\tau \) satisfy \( \|\xi_n(s) - x_n(s)\| \leq \epsilon \) for all \( s \in \mathbb{N} \) when the actions \( \sigma_n(0)\sigma_n(1) \ldots \) are implemented for both systems. For a set \( A \) we have,
\[
\kappa_\eta(A \oplus B_\infty(0, \epsilon)) \subseteq \{x \in A : \kappa_\eta(x) = x\} \subseteq A .
\]

Thus,
\[
\xi_n(s) \in \kappa_\eta(X^X_I \oplus B_\infty(0, \epsilon + \frac{\eta}{2})) \implies \xi_n(s) \in X^X_I \oplus B_\infty(0, \epsilon + \frac{\eta}{2}) \implies x_n(s) \in (X^X_I \oplus B_\infty(0, \epsilon)) \oplus B_\infty(0, \epsilon) \subseteq X^X_I ,
\]

It follows that
\[
\sum_{n \in [N]} \mathbb{I}_{G^{-\epsilon(\epsilon + \frac{\eta}{2})}}(\xi_n(s), \sigma_n(s)) \leq \sum_{n \in [N]} \mathbb{I}_{X_I}(x_n(s), \sigma_n(s)) ,
\]

thus \( \{\pi_n\}_{n \in [N]} \) is a solution also for [15]—a contradiction. \( \square \)

Example (continued). We proceed with the TCL scheduling problem by adjusting the constraints. The mode-counting constraints [4] lack a state space part, and are therefore not
are then\( \geq \) which ensures the continued positivity of the states: \( R \) and (state-dependent) admissible control set \( W \).

In the following, we use the compact notation \( \{ q \in Q \} \) aggregate states labeled \( q \) that describe the number of individual systems that are at state \( q \). By also introducing control actions \( r_q^\mu \) that represent the number of systems at state \( q \) using action \( \mu \), the aggregate dynamics can be written as

\[
\dot{w}_q(s + 1) = \sum_{\mu \in U} \sum_{q' \in N_q^\mu} r_{q'}^\mu(s), \quad q \in Q, \tag{17}
\]

where \( N_q^\mu \) is the set of predecessors of \( q \) under the action \( \mu \):

\[
N_q^\mu = \left\{ q' \in Q : q' \xrightarrow{\mu} q \right\}.
\]

We constrain the control actions \( r_q^\mu \) such that for all \( \mu \in U \),

\[
\forall q \in Q, \quad \sum_{\mu \in U} r_q^\mu(s) = w_q(s), \tag{18}
\]

which ensures the continued positivity of the states: \( w_q(s + 1) \geq 0 \) for \( q \in Q \). Furthermore, the invariant \( ||w(s)||_1 = N \) holds over time, where \( N \) is the total number of subsystems. In the following, we use the compact notation

\[
\Gamma : w(s + 1) = Br(s),
\]

to denote this system, where \( B \) is composed of the incidence matrices \( A^\mu, \mu \in U \) of the system graph. The state space \( W \) and (state-dependent) admissible control set \( R \) of this system are then

\[
W = \left\{ w \in \mathbb{N}^{|Q|} : ||w||_1 = N \right\},
\]

\[
R(w) = \left\{ r \in \mathbb{N}^{|Q||U|} : (18) \text{ holds for } (w, r) \right\}.
\]

B. Graph Properties

Next we connect properties of the induced graph \( G = (Q, \rightarrow) \) to reachability properties of the aggregate dynamics.

We first define a concept of controllability on a subset of nodes \( D \subset Q \) for the aggregate dynamics \( \Gamma \). Similarly as for controllability of linear systems on a subspace, controllability of \( \Gamma \) on \( D \) means that the system can be steered between any two aggregate states with support on \( D \).

**Definition 4.** A subset of nodes \( D \subset Q \) is completely controllable for \( \Gamma \) if for any two states \( w, w' \) with support on \( D \) such that \( ||w||_1 = ||w'||_1 \), there exists a finite horizon \( T \), states \( \{ w(s) \}_{s=0}^T \), and controls \( \{ r(s) \}_{s=0}^{T-1} \) satisfying (18), such that \( w(0) = w, w(T) = w', \) and \( w(s + 1) = Br(s) \) for \( s \in [T] \).

**Theorem 3.** If a strongly connected component \( D \) is aperiodic, it is completely controllable for \( \Gamma \).

**Proof.** It is known that the incidence matrix \( A_D \) of an aperiodic, strongly connected graph \( D \) is primitive \( \mathbb{N} \) i.e., there exists an integer \( T \) such that all entries of \( A_D^T \) are positive. This means that for each node pair \( (q_j, q_k) \), there exists a path of length \( T \) that connects them. Thus, by sending \( p_j \) systems along paths \( q_j \rightarrow q_k \) such that \( \sum_j p_j = w_j \) and \( \sum_j p_j = w_k \), the state at time \( T \) is equal to \( w' \). We can define aggregate controls \( r(s) \) that realize these paths by switching the correct number of systems at each node over time.

In the case of periodicity, it is not possible to reach every state since the parity structure of the initial state is preserved along the trajectories. However, within this restriction, the system is still controllable in a certain sense. If a strongly connected component \( D \) has period \( P \), its nodes can be labeled with a function \( L_P : D \rightarrow [P] \) such that a node \( q_1 \) with \( L_P(q_1) = p \) only has edges to nodes \( q_2 \) with \( L_P(q_2) = (p+1) \mod P \). Let \( D_0, \ldots, D_{P-1} \) be the subsets of nodes induced by the equivalence relation \( q_1 \sim q_2 \iff L_P(q_1) = L_P(q_2) \).

**Corollary 1.** The subsets of nodes \( D_p \) for \( p \in [P] \) as constructed above are completely controllable for \( \Gamma \).

**Proof.** We can connect the nodes in \( D_p \) with edges that correspond to paths of length \( P \) in \( D \). By construction, the resulting graphs are aperiodic, so the previous result applies.

It is well-known that if a discrete-time linear system is completely controllable, its reachable set for a time \( s \geq n \), where \( n \) is the system dimension, is the entire state-space; otherwise, it is an affine subspace that depends on \( s \) and the initial state. The preceding results show a corresponding result for the aggregate dynamics \( \Gamma \)—a linear system with input constraints evolving on an integer lattice. The controllability in this setting is entirely characterized by the properties of the graph representing the abstraction. For the controllable case (aperiodic graph), the reachable set of \( \Gamma \) at a time \( s \geq T \), where \( T \) is the controllability horizon from the proof of Theorem 3, is the set of all positive integer-valued vectors satisfying \( ||w||_1 = N \). In case of periodicity with a period \( P \),

\(^2\)A state \( w \) having support on \( D \) means that \( w_q = 0 \) for \( q \notin D \).
it is the intersection of this lattice set with an affine subspace that depends on \((s \mod P)\) and the parity structure of the initial state.

C. Graph assignments

We concentrate on solution trajectories that consist of a finite prefix phase and a periodic suffix phase defined on cycles. The main idea is to steer the subsystems to cycles during the prefix part, and to let them follow periodic trajectories over the cycles in the suffix part. Controllability of the aggregate dynamics plays a role in what type of periodic trajectories the subsystems can be steered to. On the other hand, periodic trajectories help us guarantee that the counting constraints in the deterministic finite counting problem are satisfied over an infinite time horizon, by analyzing a finite suffix.

We consider cycles \(C\) of the form
\[
C = (q_0, \mu_0, q_1)(q_1, \mu_1, q_2) \ldots (q_{t-1}, \mu_{t-1}, q_0),
\]
(19)
of length given by \(|C| = I\), where \((q_i, \mu_i, q_{i+1}) \in \rightarrow\).

In order to map from indices \(i\) in a specific cycle \(C\) to graph nodes, we define a function \(\Phi_C^q : |C| \rightarrow Q\) that for a cycle \(C\) on the form \((19)\) maps the cycle index \(i\) to the corresponding node \(q_i\) in the graph \(G\); that is, \(\Phi_C^q(i) = q_i\). Similarly, we define a function \(\Phi_C^\mu\) that specifies the outgoing node in \(C\) for cycle index \(i\); that is, \(\Phi_C^\mu(i) = \mu_i\). We next introduce the concept of a cycle assignment, which essentially denotes the “weights” (or numbers) of subsystems on each node along this cycle. If the graph represents an abstraction, the assignment counts the number of subsystems whose continuous states are in the vicinity of the abstract states on this cycle.

Definition 5. An assignment to a cycle \(C\) is a function \(\alpha : |C| \rightarrow \mathbb{R}_+\).

Definition 6. An integer assignment to a cycle \(C\) is an assignment \(\alpha\) to \(C\) such that \(\alpha(i) \in \mathbb{N}\) for \(i \in |C|\).

For a subsystem assigned to a cycle, its state circulates along the abstract states on this cycle as time progresses (provided that the appropriate actions are chosen). The movement corresponds to a circular shift of the assignment.

Definition 7. For an assignment \(\alpha\) we define its \(s\)-step circulation, denoted \(\alpha^{Cs} : |C| \rightarrow \mathbb{R}_+\) as the shifted function \(\alpha^{Cs}(i) = \alpha((i-s) \mod |C|)\).

The periodicity is manifested by the relation \(\alpha^{Cs}(|C|+s) = \alpha^{Cs}\). To capture how counting quantities vary during the circulation we introduce the following notation for the matching of a cycle and a circulated assignment.

Definition 8. For a cycle \(C\) and an assignment \(\alpha : |C| \rightarrow \mathbb{R}_+\) the \(X\)-count at time \(s\) of a循环 set \(X\) is defined as
\[
\langle C, \alpha^{Cs} \rangle_X = \sum_{i \in |C|} I_X (\Phi_C^q(i), \Phi_C^\mu(i)) \alpha^{Cs}(i).\]
(20)

If the cycle includes elements from a set \(X\) (typically coming from a counting constraint), \(20\) counts the number of subsystems contributing to the counting constraint at time \(s\), assuming that the subsystems make up the assignment \(\alpha\) at time zero and follow the cycles. The concept is illustrated in Fig. 2.

Example. For a given node \(q \in \mathbb{Q}\) and time \(s \in \mathbb{N}\), the expression
\[
\langle C, \alpha^{Cs} \rangle_{(q) \times \mathcal{U}}\]
(21)
represents the number assigned to \(q\) by the assignment \(\alpha\) when it has circulated \(s\) steps in \(C\). To illustrate, for the example in Fig. 2 (21) evaluates to the following values for \(q_0, \ldots, q_5\) and \(s = 0, \ldots, 4\):

\[
\begin{array}{cccccc}
q_0 & q_3 & q_4 & q_5 & q_2 & q_1 \\
0 & 3 & 4 & 4 & 3 & 2 \\
1 & 0 & 2 & 3 & 4 & 3 \\
2 & 0 & 3 & 2 & 3 & 4 . \\
3 & 0 & 4 & 3 & 2 & 3 \\
4 & 0 & 4 & 3 & 2 & 3
\end{array}
\]

Next we introduce functions that for given cycle-assignment pairs return the highest count over all circulations.

Definition 9. The maximal \(X\)-count for a cycle \(C\) with assignment \(\alpha\), denoted \(\Psi^X(C, \alpha)\), is the maximal number of subsystems simultaneously in \(X\) when \(\alpha\) is circulated around \(C\):
\[
\Psi^X(C, \alpha) = \max_{s \in \mathbb{N}} \langle C, \alpha^{Cs} \rangle_X.
\]

An illustration is provided in Fig. 3 for an example cycle-assignment pair. The maximal \(X\)-count can also be computed as the maximum entry in a matrix-vector product
\[
\Psi^X(C, \alpha) = \|B_C^X\alpha\|_\infty,
\]
where \(B_C^X \in \{0,1\}^{|C| \times |C|}\) is a circulant binary matrix
\[
B_C^X[i,j] = I_X (\Phi_C^q(k_{ij}), \Phi_C^\mu(k_{ij})) , k_{ij} = (i-j) \mod |C|,
\]
taking the general form
\[
B_C^X = \begin{bmatrix}
    b_0 & b_1 & \ldots & b_{|C|-1} \\
b_1 & b_2 & \ldots & b_0 \\
\vdots & \vdots & \ddots & \vdots \\
 b_{|C|-1} & b_0 & \ldots & b_{|C|-2}
\end{bmatrix}.
\]

Definition 10. The maximal joint \(X\)-count for a set of cycles \(\{C_j\}_{j \in J}\) and a matching set of assignments \(\{\alpha_j\}_{j \in J}\), denoted
\[
\begin{array}{c}
\alpha \rightarrow 2 \rightarrow 0 \rightarrow 1 \rightarrow 3 \rightarrow 3 \rightarrow 2 \rightarrow (C, \alpha)^X = 6 \\
\alpha^{C_1} \rightarrow 2 \rightarrow 0 \rightarrow 1 \rightarrow 3 \rightarrow 3 \rightarrow (C, \alpha^{C_1})^X = 3 \\
\alpha^{C_2} \rightarrow 3 \rightarrow 2 \rightarrow 2 \rightarrow 0 \rightarrow 1 \rightarrow 3 \rightarrow (C, \alpha^{C_2})^X = 5 \\
\alpha^{C_3} \rightarrow 3 \rightarrow 2 \rightarrow 2 \rightarrow 0 \rightarrow 1 \rightarrow 3 \rightarrow (C, \alpha^{C_3})^X = 7 \\
\end{array}
\]

Fig. 2. The diagram represents a cycle \(C\) comprising nodes \(q_0, q_1, \ldots, q_5\) matched with a circulating assignment \(\alpha = [2, 0, 1, 3, 3, 2]\). The right column shows how the \(X\)-count varies as the assignment \(\alpha\) circulates for a counting set \(X \supset \{q_0, q_2, q_1\}\), i.e. matching the cycle indices 0, 2, and 3.
Definition 11. A control strategy for an aggregate initial state \( w(0) \) is of prefix-suffix type if it consists of a finite number of inputs \( r(0), \ldots, r(T-1) \), and a set of cycles \( \{C_j\}_{j \in J} \) with assignments \( \{\alpha_j\}_{j \in J} \) such that the cycles are populated with their respective cycle assignments at time \( T \).

Given aggregate initial states \( w(0) \), a set \( \{C_j\}_{j \in J} \) of cycles in \( G \), and a prefix horizon \( T \), prefix-suffix solutions can be extracted from feasible points of the following linear feasibility problem:

\[
\begin{align*}
\text{find } & \alpha_0, \ldots, \alpha_{J-1} (\text{cycle assignments}), \\
& r(0), \ldots, r(T-1) (\text{aggregate inputs}), \\
& w(1), \ldots, w(T) (\text{aggregate states}), \\
\text{s.t. } & \sum_{q \in Q} \sum_{\mu \in \mathcal{U}} 1_{x_1}(q, \mu) r_q^\mu(s) \leq R_l, \ s \in [T], l \in [L], \ (25a) \\
& \Psi^X(\{C_j\}_{j \in J}, \{\alpha_j\}_{j \in J}) \leq R_l, \ l \in [L], \ (25b) \\
& w(s+1) = Br(s), \ \ s \in [T], \ (25c) \\
& w_q(T) = \sum_{j \in [J]} \langle C_j, \alpha_j \rangle_{\{q\} \times \mathcal{U}}, \ \ q \in Q, \ (25d) \\
& \sum_{\mu \in \mathcal{U}} r_q^\mu(s) = w_q(s), \ \ s \in [T], q \in Q, \ (25e) \\
& r_q^\mu(s) \geq 0, \ \ s \in [T], q \in Q, \mu \in \mathcal{U}. \ (25f)
\end{align*}
\]

The number of variables and (in)equalities in (25) is

\[
O(T|Q||\mathcal{U}| + \sum_{j \in [J]} |C_j|),
\]

respectively (not counting positivity constraints that solvers handle easily). Crucially, these numbers do not depend on the total number of subsystems \( N \) which makes the approach suitable for large \( N \) and moderate-sized graphs.

Secondly, the following result drastically reduces the number of inequalities in (25b) representing suffix counting constraints.

Proposition 3. The joint \( X \)-count for two cycles \( C_0 \) and \( C_1 \) with co-prime length, i.e. \( \gcd(|C_0|, |C_1|) = 1 \), can be computed as

\[
\Psi^X(\{C_0, C_1\}, \{\alpha_0, \alpha_1\}) = \Psi^X(C_0, \alpha_0) + \Psi^X(C_1, \alpha_1).
\]
Proof. If \( |C_0| \) and \( |C_1| \) are coprime, by the Chinese Remainder Theorem, the equations \( k_1 = s \mod |C_1| \) and \( k_2 = s \mod |C_2| \) have a unique solution \( s < |C_0| \) for every pair \( k_1 < |C_1|, k_2 < |C_2| \). It follows that every circulation of \( \alpha_0 \) in \( C_0 \) at some point coincides with every circulation of \( \alpha_1 \) in \( C_1 \); hence the maximal joint \( X \)-count is equal to the sum of individual \( X \)-counts for the two cycles. \( \square \)

Therefore, for sets of cycles \( \{C_j\}_{j \in J} \) and \( \{C_j'\}_{j' \in J'} \) with the property that \( \text{gcd}(|C_j|, |C_j'|) = 1 \) for all pairs \( j \in J, j' \in J' \), it holds that

\[
\Psi^X(\{C_j\}_{j \in J \cup J'}, \{\alpha_j\}_{j \in J \cup J'}) = \Psi^X(\{C_j\}_{j \in J'}, \{\alpha_j\}_{j \in J'}) + \Psi^X(\{C_j\}_{j \in J}, \{\alpha_j\}_{j \in J}).
\]

(26)

Thus, if the cycle set can be partitioned into sets of cycles with mutually co-prime length, the number of inequalities can be reduced.

Example. To exemplify this reduction, consider a set of cycles with lengths ranging from 2 to 20. We have

\[
\text{lcm}(21) = 232792560,
\]

\[
\text{lcm}(21) \setminus \{11, 13, 17, 19\} + 11 + 13 + 17 + 19 = 5100.
\]

The number 232792560 is the number of inequalities in the naïve approach (22), which by (26) can be drastically reduced to 5100 if the cycles of prime lengths 11, 13, 17, and 19 are considered separately.

If the number of constraints is still prohibitively large, it can be replaced by a conservative constraint as follows.

Remark 2. The constraint (25b) can be substituted by the conservative constraint

\[
\sum_{i \in N} \Psi^X_i(\{C_j\}_{j:|C_j|=i}, \{\alpha_j\}_{j:|C_j|=i}) \leq R_i, \quad \forall i \in [L],
\]

which groups cycles by cycle length \( i \) and disregards effects from periodicity. The number of constraints in this case becomes \( L(1 + \|\bigcup_{j \in J} \{C_j\}\|_1) \) instead of \( L \text{lcm}(\{C_j\}_{j \in J}) \), where \( \bigcup_{j \in J} \{C_j\} \) is the set of cycle lengths without repetition.

While the total number of subsystems \( N \) does not impact the number of inequalities or constraints, it might affect the performance of integer linear program solvers since the number of candidate integer points grows with \( N \). In addition, the converse results in Section VI depend on \( N \)—in order to prove infeasibility of the problem very large horizons \( T \) and/or cycle sets may be required. If \( N \) is prohibitively large for this purpose it can be scaled down if a certain divisibility condition holds: if there is a common divisor \( S \) that divides \( w_q(0) \) for all \( q \in Q \), and that divides \( R_l \) for all \( l \in [L] \), then there is a 1-1 correspondence between solutions of (25) and solutions of its analogue obtained from the substitutions \( w(0) \rightarrow w(0)/S \) and \( R_l \rightarrow R_l/S \). The correspondence simply consists in scaling \( r \), \( w \), and the assignments \( \alpha \) with the same \( S \).

F. Control strategy extraction

We conclude the section by giving a switching protocol that solves the instance (23) from a feasible solution of (25).

Algorithm 1: Switching protocol.

Data: Time \( s \), current state \( \xi_n(s) \) for \( n \in [N] \)

Result: Switching signals \( \sigma_n(s) \) for \( n \in [N] \)

1 if \( s < T \) then

2 Select \( \sigma_n(s) \) s.t. for all \( q \in Q, \mu \in U; \)

\[
\sum_{n \in [N]} 1_{\{q, \mu\}}(\xi_n(s), \sigma_n(s)) = r_q^w(s);
\]

else

4 Select \( \sigma_n(s) \) s.t. for all \( q \in Q, \mu \in U; \)

\[
\sum_{n \in [N]} 1_{\{q, \mu\}}(\xi_n(s), \sigma_n(s)) = \psi_n(s);
\]

5 end

Theorem 4. If \( \{r(s)\}_{s \in [T]}, \{w(s)\}_{s \in [T+1]}, \{\alpha_j\}_{j \in [J]} \) is a feasible solution of (25), then input selection according to the switching protocol in Algorithm 1 is recursively feasible, and solves the instance (23).

Remark 3. The switching protocol in Algorithm 1 is memoryless; input construction depends only on the current states \( \{\xi_n(s)\}_{n \in [N]} \) and on auxiliary information from the solution of (25). However, for implementation central coordination is required at each time step. The coordination requirement can be relaxed by simulating the system up to time \( T \) and assigning an individual prefix and suffix to each subsystem. Then decentralized open-loop prefix and suffix controllers can be constructed that realize these individual prefix-suffix paths and mimic the performance of the centralized protocol without communication requirements.

VI. ANALYSIS OF THE PROPOSED LINEAR PROGRAM

Theorem 4 establishes that solving (25) provides a correct solution to a deterministic finite instance of Problem 1. We now discuss completeness of the solution approach and specify the information that can be obtained from (in)feasibility of (25) when it is solved as an integer linear program, and when integer constraints are relaxed to obtain more efficiently solvable linear programs. Table II summarizes the results of this section.

A. Converse Results

The first result states that the restriction to prefix-suffix form is without loss of generality, provided that the prefix horizon is sufficiently large, and that the suffix cycle set is sufficiently rich.

Theorem 5. Suppose that there is a solution to the instance (23). Then there is a feasible solution to (25) with a prefix length \( T \) of at most \( (\|Q\| N - 1) \) and a suffix consisting of cycles of length at most \( (\|Q\| N - 1) \).

The number of cycles up to a given length can be very large; next we present a result that restricts the analysis to a much smaller set. The key observation is that an assignment can be “averaged” over its cycle without violating any counting bounds. The averaging idea is illustrated in Fig. 4 and captured in the following definition.
Given feasibility or infeasibility of (25) in various configurations, this table lists the inferences that can be made according to results in Section VI.

| LP/ILP | Feasible? | Cycle set | T | Result | Why? |
|--------|-----------|-----------|---|--------|------|
| ILP    | Yes       | Any       | Any | Solution | Theorem 4 |
| LP     | Yes       | Any       | Any | Approximate solution | Theorem 7/8 |
| ILP    | No        | \( \{ C : |C| \leq |Q| \binom{|Q|+N-1}{N} \} \) | \( (|Q|+N-1) \frac{(\text{diam}(G)^2+1)N}{\epsilon} \) | No solution exists | Theorem 5 |
| LP     | No with \( \epsilon \)-relaxation | \( \{ C : C \text{ simple} \} \) | \( (|Q|+N-1) \frac{(\text{diam}(G)^2+1)N}{\epsilon} \) | No solution exists | Theorem 6 |

![Fig. 4. Illustration of a non-average assignment \( \alpha : [6] \to \mathbb{N} \), and three averaged assignment with periods 1 (aperiodic), 2, and 3. All assignments have total weight 12, i.e., \( ||\alpha||_1 = ||\alpha_1||_1 = ||\alpha_{\{3,9\}}||_1 = ||\alpha_{\{2,4,6\}}||_1 = 12 \). Note that average assignments are not necessarily integral.](image)

**Definition 12.** For a cycle \( C \), a graph period \( P \) dividing \( |C| \), and total weights \( N_0, \ldots, N_{P-1} \), the \( P \)-average assignment \( \bar{\alpha}_{\{N_p\}_{p \in [P]}} \) is defined as

\[
\bar{\alpha}_{\{N_p\}_{p \in [P]}}(i) = \frac{N_{i \mod P}}{|C|/P}, \quad \forall i \in [|C|].
\]

In the case \( P = 1 \), this assignment has a constant \( X \)-count for any cycle, more precisely:

\[
\langle C, \bar{\alpha}_{\{N_0\}} \rangle^X = \frac{N_0}{|C|} \langle C, 1 \rangle^X
\]

for all \( s \), where \( \langle C, 1 \rangle^X \) simply counts the number of node-action pairs in \( C \) that are in the counting set \( X \). As a consequence, for any assignment \( \alpha \), it holds that

\[
\Psi^X \langle C, \alpha \|\|_1 \rangle \leq \Psi^X \langle C, \alpha \rangle.
\]

In other words, if the averaged assignment for a given total weight does not satisfy counting bounds, no assignment does.

A more general result (Lemma 7 in the appendix) shows that a cyclic suffix can be mapped onto a (possibly non-integer) suffix defined on simple cycles via averaging. It turns out that these averaged assignments can be reached by an infinitesimal relaxation of the counting bounds since they preserve the parity structure of the initial condition. The reachability results in Section V-A do not take counting constraints into account; when such constraints are present they may conflict with reachability. Nevertheless, by introducing an arbitrarily small relaxation of the counting constraints we can ensure that reachability is preserved. The magnitude of the relaxation does however impact the worst-case time required to control the aggregate system to a new state.

**Theorem 6.** Suppose that there exists an integer solution to the instance \( (N, \Sigma_{\text{DFTS}}, \{\alpha_n(0)\}_{n \in [N]}, \{X_i, R_i\}_{i \in [L]}) \). Let \( \text{diam}(G) \) be the diameter of the induced graph. Then, if every counting constraint \( (X_i, R_i) \) is relaxed with an absolute factor \( \epsilon \) to \( (X_i, R_i + \epsilon) \), the non-integer version of the linear program (25) with prefix horizon \( \frac{(\text{diam}(G)^2+1)N}{\epsilon} \) and the cycle set consisting of all simple cycles, is feasible.

**B. Rounding of Non-Integer Solution**

If the linear program (25) is too large to be solvable as an integer program it may still be possible to solve it using a standard LP solver and round the result to obtain an integer solution. One option is to use a probabilistic discrepancy-minimizing rounding procedure (e.g. (22)) which allows specified relationships to be preserved after rounding; thus introducing a counting constraint violation but maintaining the validity of the solution (e.g. dynamics, prefix-suffix connection). Here we instead propose a heuristic to round only the suffix part of the solution and analyze its performance under certain assumptions on cycle structure. For a given integer suffix, the prefix part of (25) can be solved to find a matching prefix—a problem that is typically much smaller.

Counting constraints may be violated as a result of the rounding and we give bounds for the magnitude of the worst-case violation. Given an aperiodic graph and (non-integer) cycle-assignment pairs \( \{C_j, \alpha_j\}_{j \in [J]} \) that satisfy the counting constraints, we propose the following rounding procedure:

1. Assign an integer number of subsystems to each cycle that is close to the original weight, i.e., we want to find integers \( N_j \) s.t. \( \sum_{j \in [J]} N_j = \sum_{j \in [J]} \|\alpha_j\|_1 \) and s.t. \( N_j \) is close to \( \|\alpha_j\|_1 \). This can easily be achieved in a way s.t. \( |N_j - \|\alpha_j\|_1| \leq 1 \).

2. Find individual integer assignments with total weight \( N_j \) that are close to the average assignments, i.e., we want to find integer assignments \( \tilde{\alpha}_j \) s.t. \( \|\tilde{\alpha}_j\|_1 = N_j \) and s.t. \( \alpha_j \) is close to \( \tilde{\alpha}_j \).

To this end, we let \( \kappa_1 \) and \( \kappa_2 \) be the quotient and remainder when dividing \( N_j \) by \( |C_j| \), i.e., \( \kappa_1 = \lfloor N_j/|C_j| \rfloor \) and \( \kappa_2 = N_j \mod |C_j| \). Then let \( d = |C_j|/\kappa_2 \). We consider the pseudo-periodic assignment \( \tilde{\alpha}_j \) defined as follows:

\[
\tilde{\alpha}_j(i) = \kappa_1 + 1, \quad \text{for } i \in \{dk\}_{k \in [\kappa_2]},
\]

\[
\tilde{\alpha}_j(i) = \kappa_1, \quad \text{otherwise}.
\]

This assignment is pseudo-periodic in the sense that the 1’s are evenly distributed with distance \( d \) before they are rounded to integer points, as illustrated in Fig. 5. It is easy to see that \( \|\tilde{\alpha}_j\|_1 = N_j \).

3. Find a prefix using (25) that steers to \( \{C_j, \tilde{\alpha}_j\}_{j \in [J]} \).

If the graph is periodic the rounding can be done so as to preserve the parity structure of the original solution and
guarantee reachability, but the details are omitted here. Assuming aperiodicity, we want to compute the counting bounds for the rounded solution with the counting bounds for the original solution. First we give a result that assumes a certain structure of a cycle, namely that all nodes that contribute to the counting set $X$ are placed in sequence in the cycle. Such structure is often present in practical examples as connected counting regions tend to lead to consecutive parts in cycles.

**Proposition 4.** Let $R_C^X = \langle C, 1 \rangle^X$ be the number of nodes in a cycle $C$ that contribute to $X$-counting. If all such nodes are consecutive, i.e., $(q_i, \mu_i) \in X$ for $i \in [R_C^X]$ and $(q_j, \mu_j) \notin X$ for $j \notin [\lceil C \rceil \setminus R_C^X]$, then the rounding procedure (29) satisfies

$$
\Psi^X(C_j, \alpha_j) \leq \Psi^X(C_j, \alpha_{N_j}) + 1.
$$

**Proof.** When the assignment $\alpha_j$ circulates in $C_j$, exactly $R_C^X$ contiguous indices of $\alpha_j$ contribute to the $X$-count. We bound the number of contributing indices with value $k_1 + 1$. Let $[i_0, i_0 + R_C^X - 1]$ be a (circular) sequence representing $R_C^X$ contributing indices. Consider Fig. 5 any point that ends up in the sequence after left-rounding must satisfy $dk \in [i_0, i_0 + R_C^X]$, where $k \in [\kappa_2]$. Since each left-closed, right-open interval of length $d$ captures exactly one point of the form $dk$, there are at most $\lceil R_C^X/d \rceil$ such points. Therefore by (27),

$$
\Psi^X(C_j, \alpha_j) \leq k_1 R_C^X + \left\lceil \frac{R_C^X}{d} \right\rceil \leq k_1 R_C^X + \frac{R_C^X}{d} + 1
$$

$$
\leq \frac{R_C^X}{|C_j|} (k_1 |C_j| + \kappa_2 + 1) = \Psi^X(C_j, \alpha_{N_j}) + 1.
$$

**Corollary 2.** If $C_j$ has at most $p$ $X$-segments (consecutive nodes taking values in $X$), the rounding (29) satisfies

$$
\Psi^X(C_j, \alpha_j) \leq \Psi^X(C_j, \alpha_{N_j}) + p.
$$

**Proof.** Follows from applying Proposition 4 to each of the $p$ segments.

We now incorporate these results into a bound on the counting constraint violation in the overall rounding procedure. Again, this bound does not depend on the total number of subsystems $N$. The difference between the original counting bounds and their relaxed counterparts therefore becomes insignificant as $N$ grows.

**Theorem 7.** For a given counting constraint $(X, R)$ satisfied by $\{\alpha_j\}_{j \in [J]}$, the following bound holds for the overall suffix $j$-rounding procedure

$$
\sum_{j \in [J]} \Psi^X(C_j, \alpha_j) \leq R + J + \sum_{j \in [J]} p_j^X,
$$

where $p_j^X$ is the number of $X$-segments in the $j$-th cycle. Thus the relaxed counting constraint $(X, R + J + \sum_{j \in [J]} p_j^X)$ is guaranteed to be satisfied by the rounded suffix $\{\alpha_j\}_{j \in [J]}$.

**Proof.** By the rounding procedure, Corollary 2 and (28),

$$
\Psi^X(C_j, \alpha_j) \leq \Psi^X(C_j, \alpha_{N_j}) + p_j^X
$$

$$
\leq \Psi^X(C_j, \alpha_{|\alpha_j|}) + 1 + p_j^X \leq \Psi^X(C_j, \alpha_j) + 1 + p_j^X.
$$

Summing over $j \in [J]$ gives the result.

If the structure required in Proposition 4 is not present, the following is a worst-case bound on the counting constraint violation due to rounding.

**Theorem 8.** The rounding (29) satisfies

$$
\Psi^X(C_j, \alpha_j) \leq \Psi^X(C_j, \alpha_{N_j}) + \left\lceil \frac{|C_j|}{4} \right\rceil.
$$

**Proof.** We assume that $N_j < |C_j|$, since any multiple of $|C_j|$ can be assigned as the average assignment. The number of nodes contributing to the $X$-count is upper bounded by $\min(R_C^X, N_j)$, hence,

$$
\Psi^X(C_j, \alpha_j) - \Psi^X(C_j, \alpha_{N_j}) \leq \min(R_C^X, N_j) - \frac{N_j |C_j|}{R_C^X}
$$

$$
= |C_j| \min \left( \frac{R_C^X}{|C_j|} \left( 1 - \frac{N_j}{|C_j|} \right), \frac{N_j}{|C_j|} \left( 1 - \frac{R_C^X}{|C_j|} \right) \right) \leq \frac{|C_j|}{4},
$$

where the last step follows from $\max(a, b) \leq \min(ab, (1-a)(1-b)) = 1/4$.

These rounding bounds can be precomputed (using all cycles used in the LP instead of all cycles with non-zero assignments) and the counting constraints can be strengthened accordingly to ensure that the original constraints are satisfied after rounding.

**VII. EXTENSION TO STRONG HETEROGENEITY**

Above we considered mild heterogeneity in the continuous dynamics that allowed us to construct a single abstraction that captures all the possible behaviors. If there is significant heterogeneity in the collection of subsystems, this may no longer be possible while maintaining a good approximation.

To alleviate this shortcoming, we can extend our synthesis method to a multi-class setting where each subsystem belongs to a particular class, and each class has only mild heterogeneity among its members. One abstraction per class can then be constructed, and the counting problem can be solved jointly for the different classes. In addition, counting constraints can be extended to capture class identity. For instance, we can posit that at least $R_1$ subsystems of class 1 be present in a given area, or guarantee that no more than $R_2$ subsystems of class 2 are in a particular mode.

To formalize these ideas, consider $H$ transition systems $\Sigma_h = (Q^h, \delta^h, \longrightarrow^h, Y^h)$ for $h \in [H]$. Then the multi-class counting problem is as follows:

**Problem 2.** Consider $N$ subsystems divided in $H$ classes such that class $h$ has $N_h$ members, and $\sum_{h \in [H]} N_h = N$. Subsystems in class $h$ are governed by the transition system
Given $L$ counting constraints $\{\prod_{h \in [H]} X_h^{(h), i} R_i \} i \in [L]$ with counting sets $X_h^{(h)} \subset \mathbb{Q}^h \times I^h$, synthesize individual switching protocols $\{\pi_n^{(h)}\}_{n \in [N_h]}$ such that the generated actions $\sigma_n^{(h)}(0) \sigma_n^{(h)}(1) \sigma_n^{(h)}(2) \cdots$ and trajectories $X_h^{(h)}(0) X_h^{(h)}(1) X_h^{(h)}(2) \cdots$ satisfy the counting constraints

$$
\sum_{h \in [H]} \sum_{n \in [N_h]} I_{X_h^{(h)}}(x_h^{(h)}(s), \sigma_n^{(h)}(s)) \leq R_l, \quad \forall s \in \mathbb{N}, \forall l \in [L].
$$

The linear program (25) can easily be extended to the multi-class setting, at the cost of additional variables and constraints. Assuming similar abstraction parameters and cycle selections, a problem with two classes has roughly twice as many variables as a problem with a single class. The next section includes an example showcasing how a multi-class counting problem can account for parameter heterogeneity by constructing two abstractions for a family of continuous-time systems.

### VIII. Examples

We showcase the method on two examples: one numerical example and the TCL scheduling problem. The examples are computed with our prototype implementation available at https://github.com/pettni/mode-count, which uses Gurobi as the underlying ILP solver.

#### A. Numerical Example

Our first example is the following non-linear system:

$$
\frac{dx_1}{dt} = -2(x_1 - u) + x_2, \\
\frac{dx_2}{dt} = -(x_1 - u) - 2x_2 - x_2.
$$

(30)

It can be shown that for a constant $u$ the system is incrementally stable and that the $K_L$-function

$$
\beta(r, t) = \sqrt{2r} \left\| \exp \left( \begin{bmatrix} -2 & 1 \\ -1 & -2 \end{bmatrix} t \right) \right\|_2
$$

satisfies (2). We consider two modes $\mu_1$ and $\mu_2$ corresponding to the constant inputs $u = -1$ and $u = 1$.

We consider the domain $X = \{(x_1, x_2) : x_1 \in [-2, 2], x_2 \in [-1.5, 1.5]\}$. For a large $N$, we introduce mode-counting constraints $(X_{\mu_1}, 0.55N)$ and $(X_{\mu_2}, 0.55N)$ for $X_{\mu} = X \times \{\sigma\}$ stating that at most 55% of the subsystems can use the same dynamical mode at any given time. In addition we consider a balancing objective; namely that no more than 55% of the subsystems should be in one of the sets $X_1 = \{(x_1, x_2) : x_1 \geq 0\}$ or $X_2 = \{(x_1, x_2) : x_1 \leq 0\}$; expressed by the counting constraints $(X_{1, 0.55N})$ and $(X_{2, 0.55N})$. Furthermore, we want all subsystems to repeatedly visit these two sets.

Using abstraction parameters $\eta = 0.05$, $\tau = 0.32$ we obtain an abstraction that is 0.1-approximately bisimilar to the time-discretization of (30). In order to guarantee that the counting constraints are satisfied, we therefore need to expand the counting sets as $X_1 = \{(x_1, x_2) : x_1 \geq -0.1\}$ and $X_2 = \{(x_1, x_2) : x_1 \leq 0.1\}$. We proceed by solving the discrete counting problem with randomized initial conditions and a horizon $T = 10$. We sampled 100 randomized cycles that visit both $X_{\mu_1}$ and $X_{\mu_2}$, in order to achieve the second objective. We solved the problem for $N = 10^k$ for $k = 2, \ldots, 9$; the solving times are shown in Table II and illustrate that the difficulty of this problem is largely independent of $N$.

| $N$ | $10^2$ | $10^3$ | $10^4$ | $10^5$ | $10^6$ | $10^7$ | $10^8$ | $10^9$ |
|-----|--------|--------|--------|--------|--------|--------|--------|--------|
| Time (s) | 7.8 | 11.1 | 12.0 | 12.8 | 11.5 | 13.2 | 13.0 | 11.0 |

Fig. 6 illustrates the number of systems that are in the counting sets over time in a trajectory, and Fig. 7 demonstrates some of the cycles that make up the suffix part of the solution.

#### B. Application Example: TCL Scheduling

We use the following model for the dynamics of the temperature $x_n$ of an individual TCL [10]:

$$
\frac{dx_n(t)}{dt} = -a_n (x_n(t) - \theta_n^o) - b_n P_m n(t) \sigma_n(t).$$

(31)

We assume that there are two distinct populations of TCLs, i.e., that all sets of parameters $(a_n, b_n, \theta_n^o, P_m n)$ are $\delta$-close to one of two nominal parameter configurations. Each nominal parameter configuration represents a mildly heterogeneous class (c.f. Section VII). The parameter values for the nominal configurations are listed in Table III along with the abstraction parameters $\eta$ and $\tau$ used for each class and the allowed deviation $\delta$ from these nominal values.
TABLE III
PARAMETER VALUES FOR THE TWO CLASSES OF SUBSYSTEMS. NOTE THAT THE TIME DISCRETIZATIONS NEED TO BE IDENTICAL FOR CONCURRENT EXECUTION.

| Parameter                  | Class 1                  | Class 2                  |
|----------------------------|--------------------------|--------------------------|
| Nominal \([a, b, \theta_n, P_n]\) | \([2, 2, 3, 2, 5, 6]\)  | \([2, 2, 2, 2, 3, 2, 5, 9]\)  |
| Space discretization \(\eta\)   | 0.002                    | 0.0015                   |
| Time discretization \(\tau\)   | 0.05                     | 0.05                     |
| Error bound \(\delta\)        | 0.025                    | 0.025                    |

It can easily be shown that the \(\mathcal{KL}\)-function

\[
\beta(r, s) = re^{-sa_n},
\]

satisfies (9) with respect to (31). In addition, it satisfies the approximate bisimulation inequality (12) for an approximation level \(\epsilon = 0.2\) and using the Lipschitz constant \(a_n\) for (31); thus the results from Section IV-B apply. The constraints for the TCL problem have been introduced earlier in (4)-(5). We posit that all subsystems must remain in the temperature interval \([21.3, 23.7]\); taking the approximation into account this implies that the constraint for the discrete problem becomes \(x_n(t) \in [21.5, 23.5]\).

We randomly selected initial conditions for 10,000 systems of each class, sampled 50 random cycles for each class\(^3\) and solved the ILP (25) for a prefix length of 20 steps (corresponding to one hour). We also introduced randomized additive model errors \(\delta_n\) such that \(|\delta_n| \leq \delta\) to represent mild in-class heterogeneity. Figure 8 shows simulated trajectory densities for two different mode-on-counts, one maximal count of 6,000, i.e. \(\sum_{n \in [N]} \mathbb{I}_{\{on\}}(\sigma_n(t)) \in [0, 6000]\); and one minimal count of 6,700, i.e. \(\sum_{n \in [N]} \mathbb{I}_{\{on\}}(\sigma_n(t)) \in [6700, N]\). For comparison, the fundamental minimal upper bound is 5,595 and the fundamental maximal lower bound is 7,045 as computed from formulas in (24) that apply to a centralized full-state feedback coordinator with arbitrarily fast switching. While the objective here was not to find the maximal ranges (which could be done by adding an objective function to (25)), the fundamental limits can not be attained due to approximation errors stemming from the approximate bisimulation, incomplete cycle selection, a minimal dwell time imposed by the time discretization, etc. Figure 7 shows mode-on-counts during the same simulation for the two experiments. As can be seen, the imposed counting bounds are satisfied.

Remark 4. It would be straightforward to also introduce between-class heterogeneity in the counting constraints. Such heterogeneity could be used to model power consumption that varies between classes and thus allow a more realistic power consumption constraint to be enforced.

IX. CONCLUSION

This paper was concerned with control synthesis for very high-dimensional but permutation-symmetric systems subject to likewise symmetric counting constraints, and presented a scalable sound and (almost) complete solution to this problem. The main insight is to aggregate the individual subsystem dynamics as an integer linear system induced from an abstraction constructed for a single subsystem, thus avoiding abstraction of the whole state space. As we used an approximately bisimilar system as an abstraction, the same abstraction can represent not only identical subsystems (homogeneity) but also subsystems with slightly different parameters (i.e., almost symmetric, or mildly heterogeneous) by a slight change in the approximation factor. The control synthesis problem was then reduced to one of coordinating the number of subsystems that are in different parts of the discrete state-space of this abstraction. We characterized prefix-suffix solutions as the feasible set of an integer linear program, and showed how to interpret (in)feasibility of the program both in the integer and non-integer case.

The results were demonstrated on a TCL scheduling problem including tens of thousands of subsystems. With the proposed approach, it is possible to impose hard constraints on the overall power consumption of TCLs over an infinite

---

\(^3\)To promote diversity in the cycle sets, the cycles were selected in order to have different fractions of time in mode on.

---

Fig. 8. Density of TCLs in different parts of the temperature spectrum over time—blue parts represent regions with a larger fraction of the 20,000 subsystems. The state counting constraint (5) guarantees that no subsystem exists in the interval \([21.3, 23.7]\) which is marked with dashed lines. The first hour represents the prefix part of the solution which steers the initial state to the periodic suffix.

Fig. 9. Number of TCLs in mode on during the two simulations. As can be seen, the lower bound of 6700 is enforced for the upper (red) trajectory, while the upper bound 6000 is enforced for the lower (blue) trajectory.
time horizon, to the best of our knowledge this is a first in this domain. Counting constraints are also relevant in other application domains, including multi-agent planning and coordination as shown in [13].

Exploiting symmetries to achieve scalability in correct-by-construction methods is a promising direction and we will explore other types of symmetries in our future work. Another interesting direction is to consider other types of abstractions, including non-deterministic ones, since not all systems admit finite bisimulations. Although the idea of an aggregate system can still be used in this case, one should either solve a robust uncertain ILP, which could lead to conservative results, or consider reactive feedback solutions, for which different synthesis techniques should be developed.

ACKNOWLEDGMENT

The authors would like to thank Johanna Mathieu for insightful discussions regarding TCL coordination problem. Petter Nilsson is supported by NSF grant CNS-1239037. Necmiye Ozay is supported in part by NSF grants CNS-1446298 and ECCS-1553873, DARPA grant N66001-14-1-4045 and an Early Career Faculty grant from NASA’s Space Technology Research Grants Program.

REFERENCES

[1] C. Belta, B. Yordanov, and E. A. Gol, *Formal Methods for Discrete-Time Dynamical Systems*. Springer, 2017.
[2] P. Tabuada, *Verification and control of hybrid systems: a symbolic approach*. Springer, 2009.
[3] C. Conte, C. N. Jones, M. Morari, and M. N. Zeilinger, “Distributed synthesis and stability of cooperative distributed model predictive control for linear systems,” *Automatica*, vol. 69, pp. 117–125, Jul. 2016.
[4] S. Smith, P. Nilsson, and N. Ozay, “Interdependence quantification for compositional control synthesis: An application in vehicle safety systems,” in *Proc. IEEE CDC*, 2016, pp. 5700–5707.
[5] S. Coogan and M. Arcak, “Efficient finite abstraction of mixed monotone systems,” in *Proc. HSCC*, 2015, pp. 58–67.
[6] E. S. Kim, M. Arcak, and S. A. Seshia, “Directed specifications and assumption mining for monotone dynamical systems,” in *Proc. HSCC*, 2016, pp. 21–30.
[7] A. Ghaffari, S. Moura, and M. Krstic, “Modeling, Control, and Stability Analysis of Heterogeneous Thermostatically Controlled Load Populations Using Partial Differential Equations,” *J. Dyn. Sys., Meas., Control*, vol. 137, no. 10, p. 101009, 2015.
[8] S. Koch, J. L. Mathieu, and D. S. Callaway, “Modeling and control of aggregated heterogeneous thermostatically controlled loads for ancillary services,” in *Proc. PSCC*, 2011, pp. 1–7.
[9] S. Esmaeil Zadeh Soudjani and A. Abate, “Aggregation and control of populations of thermostatically controlled loads by formal abstractions,” *IEEE Trans. Control Syst. Tech.*, vol. 23, no. 3, pp. 975–990, 2015.
[10] H. Hao, B. M. Sanandaji, K. Poolla, and T. L. Vincent, “Aggregate Flexibility of Thermostatically Controlled Loads,” *IEEE Trans. Power Syst.*, vol. 30, no. 1, pp. 189–198, 2015.
[11] R. Alur, A. Trivedi, and D. Wojtczak, “Optimal scheduling for constant-rate multi-mode systems,” in *Proc. HSCC*, 2012, pp. 75–84.
[12] R. Alur, V. Forejt, S. Mooraf, and A. Trivedi, “Safe schedulability of bounded-rate multi-mode systems,” in *Proc. HSCC*, 2013, pp. 243–252.
[13] A. Colombo and D. Del Vecchio, “Least restrictive supervisors for intersection collision avoidance: A scheduling approach,” *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1515–1527, 2015.
[14] P. Nilsson and N. Ozay, “Control synthesis for large collections of systems with mode-counting constraints,” in *Proc. HSCC*, 2016, pp. 205–214.
[15] Y. E. Sahin, P. Nilsson, and N. Ozay, “Provably-correct coordination of large collections of agents with counting temporal logic constraints,” in *Proc. ACM/IEEE ICCPS*, 2017, pp. 249–258.
[16] D. Angeli, “A Lyapunov approach to the incremental stability properties,” *IEEE Trans. Autom. Control*, vol. 47, no. 3, pp. 410–421, 2002.
[17] J. Liu and N. Ozay, “Finite abstractions with robustness margins for temporal logic-based control synthesis,” *Nonlinear Analysis: Hybrid Systems*, vol. 22, pp. 1–15, 2016.
[18] G. Pola and P. Tabuada, “Symbolic Models for Nonlinear Control Systems: Alternating Approximate Bisimulations,” *SIAM J. Control Optim.*, vol. 48, no. 2, pp. 719–733, 2009.
[19] M. W. Hirsch, S. Smale, and R. L. Devaney, *Differential Equations, Dynamical Systems, and an Introduction to Chaos*, 3rd ed. Academic Press, 2012.
[20] A. Girard and G. J. Pappas, “Approximation Metrics for Discrete and Continuous Systems,” *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 782–798, 2007.
[21] S. Lovett and R. Meka, “Constructive Discrepancy Minimization by Walking on the Edges,” *SIAM J. Comput.*, vol. 44, no. 5, pp. 1573–1582, 2015.
[22] Gurobi Optimization, Inc., "Gurobi optimizer reference manual," 2015. [Online]. Available: http://www.gurobi.com
[23] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network flows*. Prentice Hall, 1993.
[24] J. Shen, “A bound on the exponent of primitivity in terms of diameter,” *Linear Algebra and its Applications*, vol. 244, pp. 21 – 33, 1996.

APPENDIX

A. Proof of Theorem 4

We first consider the case \( s \leq T \), and claim that the selection on Line 2 is possible if

\[
 w_q(s) = \sum_{n \in \mathbb{N}} I_{(q)}(\xi_n(s)), \quad \forall q \in \mathcal{Q}. \tag{32}
\]

and, furthermore, that the selection guarantees that (32) holds at time \((s+1)\).

Due to (24), equation (32) holds at \( s = 0 \). For induction, assume that (32) holds at time \( s \). Then by (256),

\[
 \sum_{\mu \in U} \prod^\mu_q(s) = w_q(s) = \sum_{n \in \mathbb{N}} I_{(q)}(\xi_n(s)). \tag{33}
\]

The selection on line 2 amounts to for each \( q \in \mathcal{Q} \) assigning \( w_q(s) \) objects to \(|\mathcal{U}| \) “bins” such that the \( \mu \)-bin has \( \prod^\mu_q \) members; by (33) this is doable. Remark that if \( \sigma_n(s) = \mu \), then \( \xi_n(s+1) = q \) if and only if \( \xi_n(s) \in \mathcal{N}_\mu^q \). Thus,

\[
 \sum_{n \in \mathbb{N}} I_{(q)}(\xi_n(s+1)) = \sum_{n \in \mathbb{N}} \sum_{\mu \in U} I_{(q',\mu)}(\xi_n(s),\sigma_n(s))
\]

\[
 = \sum_{n \in \mathbb{N}} \sum_{\mu \in U} \sum_{q' \in U} I_{(q',\mu)}(\xi_n(s),\sigma_n(s))
\]

\[
 = \sum_{\mu \in U} \sum_{q' \in U} \prod^\mu_{q'}(s) = w_q(s+1),
\]

where the last step follows from (17). Thus (32) holds for all \( s \in [T+1] \).

Secondly, we consider the case \( s \geq T \) and claim that the selection on line 4 is possible if for all \( q \in \mathcal{Q} \)

\[
 \sum_{\mu \in U} \sum_{j \in \mathcal{U}} \bigg( C_j \alpha_j^{(s-T)} \bigg) I_{(q)}(\xi_n(s)) = \sum_{n \in \mathbb{N}} I_{(q)}(\xi_n(s)), \tag{34}
\]

and, furthermore, that the selection guarantees that (34) holds at time \((s+1)\). To show that (34) enables a selection \( \{\sigma_n(s)\}_{n \in \mathbb{N}} \) satisfying line 4, it suffices to remark that the
selection problem is equivalent to above with \( r^\mu_q(s) \) replaced by
\[
\sum_{j \in \mathbb{J}} \left< C_j, \alpha_j^<(s-T) \right>^{(q,\mu)}.
\]

Due to (25a) and (22), equation (34) holds at \( s = T \). Suppose for induction that (34) holds at time \( s \geq T \) and that a selection \( \{\sigma_n(s)\}_{n \in \mathbb{N}} \) satisfying line 4 has been made. Then,
\[
\sum_{n \in \mathbb{N}} \mathbb{1}_{\{q\}}(\xi_n(s+1)) = \sum_{n \in \mathbb{N}} \mathbb{1}_{\{q,\mu\}}(\xi_n(s+1),\sigma_n(s))
\]
\[
= \sum_{n \in \mathbb{N}} \sum_{\mu \in \mathbb{U}} \sum_{q' \in \mathbb{N}^\mu_q} \mathbb{1}_{\{(q',\mu)\}}(\xi_n(s),\sigma_n(s))
\]
\[
= \sum_{j \in \mathbb{J}} \left< C_j, \alpha_j^<(s-T) \right>^{N_q^\mu} = \sum_{j \in \mathbb{J}} \left< C_j, \alpha_j^<(s+1-T) \right>^{(q,\mu)}.
\]
The last step follows from the observation that a node \( q' \in \mathbb{N}^\mu_q \) must have a cycle index \( (i-1) \mod |C_j| \) in cycle \( C_j \), where \( i \) is the cycle index of \( q \). Thus the selection on line 4 is feasible for all \( s \geq T \).

Finally, we show that each counting constraint \((X_l,R_l)\) is satisfied. For \( s < T \) we have from line 2 and the constraint (25a):
\[
\sum_{n \in \mathbb{N}} \mathbb{1}_{X_l}(\xi_n(s),\sigma_n(s))
\]
\[
= \sum_{n \in \mathbb{N}} \sum_{\mu \in \mathbb{U}} \sum_{q \in \mathbb{Q}} \mathbb{1}_{X_l}(q,\mu) \mathbb{1}_{\{(q,\mu)\}}(\xi_n(s),\sigma_n(s))
\]
\[
= \sum_{q \in \mathbb{Q}} \sum_{\mu \in \mathbb{U}} \mathbb{1}_{X_l}(q,\mu) r^\mu_q(s) \leq R_l.
\]
Thus the counting constraints are satisfied in the prefix phase. For the suffix phase, from line 4 and (25b) it follows that
\[
\sum_{n \in \mathbb{N}} \sum_{\mu \in \mathbb{U}} \sum_{q \in \mathbb{Q}} \mathbb{1}_{X_l}(q,\mu) \mathbb{1}_{\{(q,\mu)\}}(\xi_n(s),\sigma_n(s))
\]
\[
= \sum_{q \in \mathbb{Q}} \sum_{\mu \in \mathbb{U}} \mathbb{1}_{X_l}(q,\mu) \sum_{j \in \mathbb{J}} \left< C_j, \alpha_j^<(s-T) \right>^{(q,\mu)}
\]
\[
= \sum_{j \in \mathbb{J}} \left< C_j, \alpha_j^<(s-T) \right>^{X_l} \leq R_l.
\]
Thus the switching protocol in Algorithm 1 generates inputs and trajectories that satisfy the constraints of (23).

**B. Proof of Theorem 5**

Let \( \{\pi^*_n\}_{n \in \mathbb{N}} \) be a solution to the instance \( \mathbf{Q}^c \) and consider the generated sequence of controls \( \sigma_n(s) \), \( s \in \mathbb{N} \), and aggregate states \( w_q(s) = \sum_{n \in \mathbb{N}} \mathbb{1}_{\{q\}}(\xi_n(s)) \). The number of possible values \( w(s) \) can take is finite and given by \(|\mathbb{Q}|^{|\mathbb{N}|^{s-N+1}} - \)the number of ways in which \( N \) identical objects (subsystems) can be partitioned into \( \mathbb{Q} \) sets (nodes). We can therefore find times \( T_1, T_2 \) with \( T_1 < T_2 \leq (|\mathbb{Q}|^{|\mathbb{N}|^{s-N+1}} \) such that \( w(T_1) = w(T_2) \). We show that the graph flows induced by \( \{\pi^*_n\}_{n \in \mathbb{N}} \) on the time interval \([T_1,T_2]\) can be achieved with cycle assignments: we define a flow on a graph in a higher dimension, decompose it into flows over cycles, and project the cyclic flow onto the original graph.

Let \( G = (\mathbb{Q}, \rightarrow) \) be the system graph, and define a new graph \( H = (V_H, E_H) \). The node set \( V_H = \mathbb{Q} \times \mathbb{Q} \times \ldots \times \mathbb{Q} \) contains \( T_2 - T_1 \) copies of each node in \( \mathbb{Q} \), and copies of \( q \in \mathbb{Q} \) are labeled \( q_s \) for \( s \in [T_1,T_2] \). The set of edges is defined as
\[
E_H = \{(q_s, q_{s+1}) : s \in [T_1, \ldots, T_2 - 1], (q, \tilde{q}) \in E\}
\]

An edge flow is induced on \( H \) by \( \{\pi^*_n\}_{n \in \mathbb{N}} \), obtained by letting the flow along \( \{q_s, q_{s+1}\} \) be the number of subsystems that traverses the edge \( (\tilde{q}, \tilde{q}) \in E \) at time \( s \), and by letting the flow along \( (q_{T_2}, q_{T_1}) \) be equal to the number of systems at \( q \) at time \( T_1 \). By construction, this flow is balanced at each node (i.e. inflows equal outflows).

By the flow decomposition theorem (25 Theorem 3.5), we can then find cycles and assignments in \( H \) that achieve this edge flow. As becomes evident from the proof in (25), these cycles are furthermore simple in \( H \) and thus of length at most \( |V_H| = |\mathbb{Q}|(T_2 - T_1) \). By projecting these cycles onto a single copy of \( \mathbb{Q} \), we obtain cycles and assignments in \( G \) that mimic the counting performance of \( \{\pi^*_n\}_{n \in \mathbb{N}} \) on the interval \([T_1,T_2]\) when circulated.

We can therefore define a prefix-suffix strategy by taking as the prefix part
\[
r^\mu_q(s) = \sum_{n \in \mathbb{N}} \mathbb{1}_{\{(q,\mu)\}}(\xi_n(s),\sigma_n(s)), \quad s \in [T_1],
\]
followed by a suffix part consisting of the cycles and assignments constructed as above.

**C. Proof of Theorem 6**

**Lemma 1.** Suppose \( C = C_1 \cup C_2 \) is a cycle that visits a node \( q_0 \) twice, so that it can be decomposed into two cycles \( C_1 = (q_0,q_1) \ldots (q_0,q_0) \) and \( C_2 = (q_0,q_1+1) \ldots (q_{|C_2|+1},q_0) \).

Let \( \alpha \) be an assignment to \( C \) that satisfies \( \Psi^X(C, \alpha) \leq R \), let \( P \) be the graph period, and let
\[
N_p = \sum_{i \in [|C|/P]} \alpha(p+iP), \quad p \in [P],
\]
\[
N^2_p = \frac{|C|}{|C|}N_p, \quad N^2_p = \frac{|C|}{|C|}N_p, \quad p \in [P].
\]

Then the joint \( X \)-counts for the assignment \( \tilde{\alpha}(N^2_p) \) to \( C_1 \) and the assignment \( \tilde{\alpha}(N^2_p) \) to \( C_2 \), satisfy
\[
\Psi^X(C_1, C_2, \tilde{\alpha}(N^2_p)) \leq R.
\]

**Proof of Lemma 7** We have for all \( i_1 \) that
\[
\tilde{\alpha}(N^2_p)(i_1) = \frac{PN(i_1 \mod P)}{|C|} = \frac{PN(i_1 \mod P)}{|C|} + \frac{P}{|C|} \sum_{k \in [|C|/P]} \alpha((i_1 \mod P) + kP),
\]
similarly for \( C_2 \). We note that \( P \) must divide both \(|C_1| \) and \(|C_2| \), so \((i_1 \mod |C|) \mod P = (i_1 \mod P) \) for all \( i_1 \) and similarly for \(|C_2| \). Below we use the short hand notation.
Let $\mathbf{I}_X^C(i) = \mathbf{I}_X(\Phi_0^C(i), \Phi_0^C(i))$ to indicate that the $i$'th node in a cycle $C$ is in the counting set $X$. For $j = 1, 2$ we get

$$\left\langle C_j, \tilde{\alpha}_C^0, (N_\beta^0)_{p \in \mathcal{P}} \right\rangle^X = \sum_{i \in [C_j]} \mathbf{I}_X^C(i) \left( \tilde{\alpha}_C^0(N_\beta^0)_{p \in \mathcal{P}} \right)^C \alpha(i_j - s \mod P) + kP).$$

We have $i_2 \mod P = (|C_1| + i_2) \mod P$, and $i_2 \mapsto |C_1| + i_2$ is a mapping from $i_2 \in [C_2]$ to the corresponding index in $C$. We can therefore convert a sum over both $i_1$ and $i_2$ into a sum over the index $i$ of $C$.

$$\left\langle C_1, \tilde{\alpha}_C^0, (N_\beta^0)_{p \in \mathcal{P}} \right\rangle^X + \left\langle C_2, \tilde{\alpha}_C^0, (N_\beta^0)_{p \in \mathcal{P}} \right\rangle^X = \sum_{\begin{array}{c}i \in [C_1]\end{array}} \mathbf{I}_X^C(i) \sum_{\begin{array}{c}k \in [C_1/P]\end{array}} \alpha((i - s \mod P) + kP)
\leq \max_{k \in [C_1/P]} \sum_{\begin{array}{c}i \in [C_1]\end{array}} \mathbf{I}_X^C(i) \alpha((i - s \mod P) + kP)
\leq \max_{s \in [C_1]} \sum_{\begin{array}{c}i \in [C_1]\end{array}} \mathbf{I}_X^C(i) \alpha_0^{C}(i) = \Psi^X(C, \alpha) \leq R. \quad \square$$

**Proof.** By Theorem [3] we know that a correct solution must eventually lead to periodic behavior, and Lemma [1] shows that the suffix of such a solution can be mapped into a suffix on simple cycles consisting of $P$-averaged assignments, where $P$ is the period of the graph. What remains to show is that this latter suffix is reachable from the initial state while respecting the relaxed counting constraints.

Let $w(s)$ for $s \in \mathbb{N}$ be the trajectory generated from an integer solution to (25). It satisfies the counting constraints for all $s$. From Lemma [1] we can obtain a set of simple cycles $I$ such that some $P$-averaged assignments to these cycles satisfy the counting bounds. In addition, these assignments have the same parity structure as $w(s)$, and hence as the initial condition; therefore they are reachable from the initial condition by virtue of Corollary [4]. We now propose a switching protocol to control the aggregate state to these assignments; the protocol consists in gradually steering a fraction of the systems from the original solution $w(s)$ to the $P$-averaged assignments pertaining to the cycles in $I$

We use the following notation: let $\alpha(s)$ be what remains of the correct trajectory $w(s)$, let $\beta(s)$ be the fraction currently being steered towards assignments to the simple cycles, and let $\gamma(s)$ be the fraction that has already reached these assignments. We then have $\alpha(0) = w(0)$, and $\beta(0) = \gamma(0) = 0$.

The overall system state is $\hat{w}(s) = \alpha(s) + \beta(s) + \gamma(s)$.

The protocol at time $s + 1$ is as follows.

- If $\beta(s) = 0$, set $\alpha(s + 1) = \alpha(s)(1 - \epsilon/||\alpha(s)||_1)$, and $\beta(s + 1) = \alpha(s)\epsilon/||\alpha(s)||_1$.
- If $\beta(s)$ has reached the average assignments to cycles in $I$, set $\beta(s + 1) = 0$, and $\gamma(s + 1) = \beta(s) + \gamma(s)$.
- Otherwise, steer $\beta(s)$ toward the average assignments to cycles in $I$.

We remark that the transitions are all properly connected and merely illustrate the transport of subsystem “weight” from $\alpha(s)$ via $\beta(s)$ to the average assignments represented by $\gamma(s)$. Transporting a mass $\epsilon$ takes at most time $(\text{diam}(G)^2 + 1)$ by [26] which gives a bound $T \leq (\text{diam}(G)^2 + 1)$ for the $T$ in the proof of Theorem [3]. We can thus infer that the total transportation time is upper bounded by $(\text{diam}(G)^2 + 1)N/\epsilon$, since an absolute weight $\epsilon$ is transported in each step.

We finally consider the counting bounds. By assumption, they are satisfied by $w(s)$, and by Lemma [1] they are also satisfied once the average assignments to cycles in $I$ are reached. In the meantime,

$$\tilde{w}_q^0(\epsilon(s)) = \alpha_q^0(\epsilon(s)) + \beta_q^0(\epsilon(s)) + \gamma_q^0(\epsilon(s)).$$

For every $s$, there is an integer $z \leq 1/\epsilon$ such that

$$\alpha(s) = \left(1 - \frac{\epsilon z}{N}\right)w(s), \quad \gamma(s) = \frac{\epsilon(z - 1)}{N}\gamma_0(s),$$

where $\gamma_0$ is the average assignment to the cycles. Furthermore $||\beta_q^0(\epsilon(s))||_1 \leq \epsilon$ which shows that the counts are bounded as follows:

$$\sum_{q \in Q} \sum_{\mu \in U} \mathbf{I}_X(q, \mu) \tilde{w}_q^0(\epsilon) = \sum_{q \in Q} \sum_{\mu \in U} \mathbf{I}_X(q, \mu) \left(1 - \frac{\epsilon z}{N}\right)w_q^0(\epsilon) + \epsilon(z - 1)\gamma_0(s)/N + \beta_q^0(\epsilon(s)) \leq (1 - \epsilon z/N)\beta_1 + \epsilon(z - 1)\beta_1/N + \epsilon \leq \beta_1 + \epsilon.$$

We have therefore constructed a prefix-suffix solution, where the suffix part consists of simple cycles, such that the relaxed counting bounds $(X_1, R_1 + \epsilon)$ are satisfied. \square

**Petter Nilsson** received his B.S. in Engineering Physics in 2011, and his M.S. in Optimization and Systems Theory in 2013, both from KTH Royal Institute of Technology in Stockholm, Sweden. While obtaining these degrees, he spent one year at cole Polytechnique in Paris, France, and was also a visiting researcher at California Institute of Technology in Pasadena, CA. In addition to his technical degrees, he holds a B.S. in Business and Economics from Stockholm School of Economics in Stockholm, Sweden. He is currently a Ph.D. candidate in Electrical Engineering: Systems at the University of Michigan, Ann Arbor, supervised by Jessy W. Grizzle and Necmiye Ozay. His research is focused on developing formal control algorithms for cyber-physical systems.

**Necmiye Ozay** received the B.S. degree from Bogazici University, Istanbul in 2004, the M.S. degree from the Pennsylvania State University, University Park in 2006 and the Ph.D. degree from Northeastern University, Boston in 2010, all in electrical engineering. She was a postdoctoral scholar at California Institute of Technology, Pasadena between 2010 and 2013. She is currently an assistant professor of Electrical Engineering and Computer Science, at University of Michigan, Ann Arbor. Her research interests include control of dynamical systems, optimization and formal methods with applications in cyber-physical systems, system identification, verification & validation and autonomy. She received a DARPA Young Faculty Award in 2014 and an NSF CAREER Award, a NASA Early Career Faculty Award and a DARPA Director’s Fellowship in 2016.