Noname manuscript No.
(will be inserted by the editor)

Setpoint Regulation for Stochastically Interacting Robots

Nils Napp · Samuel Burden · Eric Klavins

the date of receipt and acceptance should be inserted later

Abstract We present an integral feedback controller that regulates the average copy number of an assembly in a system of stochastically interacting robots. The mathematical model for these robots is a *tunable reaction network*, which makes this approach applicable to a large class of other systems, including ones that exhibit stochastic self-assembly at various length scales. We prove that this controller works for a range of setpoints and how to compute this range both analytically and experimentally. Finally, we demonstrate these ideas on a physical testbed.

1 Introduction

Self-assembly of complex systems and structures promises many new applications, such as easily combining different micro-fabrication technologies [1] or building arbitrary, complex nano-structures [2]. While many natural systems are reliably self-assembled at vastly different length and time scales, engineered self-assembled systems remain comparatively simple. The difficulties of engineering complex self-assembling systems are associated with large configuration spaces, our lack of understanding the relationship between local and global dynamics, and the stochastic or uncertain nature of their dynamic models.

In the context of engineering such systems, the interplay between uncertainty and sensitivity of global to local behavior can often lead to a profound lack of modularity as small unintended local interactions can drastically alter the behavior from what is expected by composition. In this paper we partially address this problem by designing a feedback controller that can regulate the expected value of the number of an arbitrary component type. This approach could be used for composition in the sense that other subsystems can rely on the presence of these regulated quantities.

We are guided by the application of *stochastic self-assembly*, in which self-assembling particles interact randomly. Such systems abound in engineered settings, such as in DNA self-assembly [2], micro and meso-scale self-assembly [1, 3–5], and robotic self-assembly [6, 7]. It is also the prevailing model for self-assembly in biological systems.

Self-assembly can be either *passive* or *active*. Designing systems that passively self-assemble is a problem of engineering a favorable free energy landscape in configuration space. Passively self-assembling systems often lack flexibility since a specific energy landscape can be difficult to adapt to new tasks. In addition, there are physical limitations to how much the energy landscape can be manipulated. The yield of a desired output structure is a function of both the shape and depth of energy wells. As a result of the physical limits on manipulating the energy landscape, passive self-assembly generally leads to low yields.

In active self-assembly, energy can be locally injected into the system. In particular, we focus on the situation where some particles have the ability to selec-

Nils Napp · Eric Klavins
Electrical Engineering
University of Washington
Seattle WA 98195
Tel.: +1 206 685 8678
Fax.: +1 206 543 3842
E-mail: nnapp@uw.edu
E-mail: klavins@uw.edu

Samuel Burden
Electrical Engineering
University of California at Berkeley
Berkeley CA 94720
E-mail: sburden@eecs.berkeley.edu

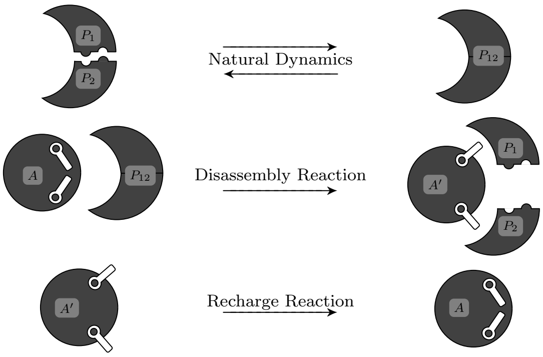

Fig. 1 Schematic representation of robot interactions. The passive robots P_1 and P_2 can form heterodimers, which can disassemble spontaneously. The active robot A can expend energy to undo bonds. When the arms of an active robot are retracted, it is charged and can actively disassemble a dimer. If the arms of an active robot are extended (denoted A') then the it is not charged, but may become charged via the recharge reaction, the rate of which can be controlled.

tively undo bonds that are formed by passive dynamics. The input of available energy is global, but how it is used to undo bonds is decided locally by the self-assembling particles, similar to the availability of nutrients that power bio-chemical processes. Active self-assembly can overcome the lack of flexibility of passively self-assembling system by making aspects of the system re-programmable while leaving other areas in the energy landscape untouched. As a result, the changes in the global dynamics remain tractable.

The particular model for active self-assembly we investigate is that of a *tunable reaction network*. We present a system of simple stochastically interacting robots that are well modeled as a tunable reaction network and demonstrate a feedback setpoint regulation scheme. Fig. 1 shows a pictorial representation of the tunable reaction network investigated in this paper. There are three robot types and several instances of each (see Fig. 2(a)). The passive robots P_1 and P_2 are able to bind and form heterodimer complexes P_{12}, which in turn can spontaneously disassemble. The active robots A can dock with heterodimers and disassemble them. The disassembly reaction leaves active robots in an uncharged state, denoted by A'. The last reaction in Fig. 1 recharges uncharged robots at a rate that is controlled externally. It corresponds to the rate at which energy is delivered to the system globally. *The control problem for this system is to regulate the number of heterodimers P_{12} in the system by adjusting the recharge rate.* (This problem is re-stated formally in Sec. 6.) While the tunable reaction network shown in Fig. 1 is comparatively simple, tunable reaction networks in general can describe much more complicated systems.

For example, many biological systems can be viewed as tunable reaction networks. Inside cells, enzymes are expressed to control the rates of various metabolic reactions. Similar to the problem solved here, one of the many functions of the biochemical processes inside cells is maintaining equilibria of chemical species. Regulating the concentration of chemical species is a particular aspect of *homeostasis*, which can be viewed as a control problem [8].

For the artificial systems depicted in Fig. 1 we propose, analyze, and implement a feedback controller. The robots serve as a physical instantiation of a tunable reaction network. Using such a simple system also allows us examine some of the model assumptions in some detail. However, the theorem and proof in Sec. 5 do not rely on any special structure of the example network, so that the results are applicable to many other systems, including ones with more species and intermediate assembly steps.

In the context of engineering self-assembling systems, the proposed feedback controller can be used to provide stable operating conditions for other self-assembling processes, much like homeostasis in biological systems. For example, in a hypothetical system with a vat of self-assembling miniature robots, we might care that the relative concentration of robot feet and robot legs is fixed in order to maximize the yield of functioning robots. In general, we envision the self-assembling systems of the future as having metabolisms of their own that regulate the various species of partially assembled objects in the system to maximize the yield of the desired final assembly.

2 Experimental Robotic Chemistry

The robots described here fall in the broad class of modular robots as there are many identical copies of each robot type comprising the overall system. For an overview of this vast area of research see, for example [9]. Specifically, the robots in this paper are stochastic modular robots as in [6, 7, 10, 11], however, they are much simpler both mechanically and electronically. Also, while many robotic platforms consist of a homogeneous group of robots, the robotic testbed described here is a heterogeneous mixture of three different robot types, Fig. 2(b)(c). The assembly of the two passive robot types P_1 and P_2 is driven by complementary shape and embedded magnets. The magnetic force creates an energy well that tends to pull P_1 and P_2 robots together to form a heterodimer. The third, active robot type can expend energy to disassemble a heterodimer into its constituents.

The energy for this disassembly is supplied to the active robots via solar panels. Each active robot stores energy from its solar panel in a capacitor, if the charge in the capacitor reaches a threshold and an active robot A is bound to a heterodimer a motor activates and disassembles it. Disassembling heterodimers depletes the on-board energy storage of active robots requiring more energy from the solar cells to disassemble additional heterodimers. Adjusting the amount of incident light changes the recharge rate of active robots and thus indirectly affects the rate at which heterodimers are disassembled.

Although this indirect approach may seem unnecessarily complicated, it possesses a key design feature that we believe justifies the added complexity: the structural, energy delivery, and computational functions reside on separate components of the overall system. We think of P_1 and P_2 as the structural components we want to control, the active robots as agents of energy delivery, and the controller implemented on a computer as the computational component. This division of labor is analogous to many biological systems where different cellular functions are largely separated into different types of molecules. We believe that such a separation of functionality in self-organization is essential to engineering large scale complex systems. Distributing the functionality in this way can yield much simpler individual components on average. For example, supplying energy externally allows micro-robots to be simpler in construction [12]. In this example, the passive robots contain no electronic components whatsoever, and the active robots only contain a simple circuit made from discrete electrical components, a motor, and a solar panel.

2.1 Physical Characteristics of Testbed

The body of each robot is machined from polyurethane prototyping foam and painted black to aid the vision system. This material is easy to machine, light, and stiff.

The robots float on an air-table shown in Fig. 2(a), which has a large HVAC blower attached to the bottom of a perforated board (blower not visible in image). The blower is able to maintain a high flow-rate of air through the table surface and allows us to float relatively heavy pieces $\approx 2.5\frac{\text{g}}{\text{cm}^2}$. The active area of the table is 60cm × 60cm. Mounted along the perimeter of the table are computer controlled solenoid valves. These valves can deliver short bursts of pressurized air from a compressor (30psi). By randomly activating these air-jets, robots on the air-table are driven to perform a random walk. The bursts are randomized and controlled via a `MATLAB` script, which also updates the state of the controller and adjust the intensity of four overhead lamps. These lamps determine the amount of incident light to the solar panels, thereby setting the recharge reaction rate.

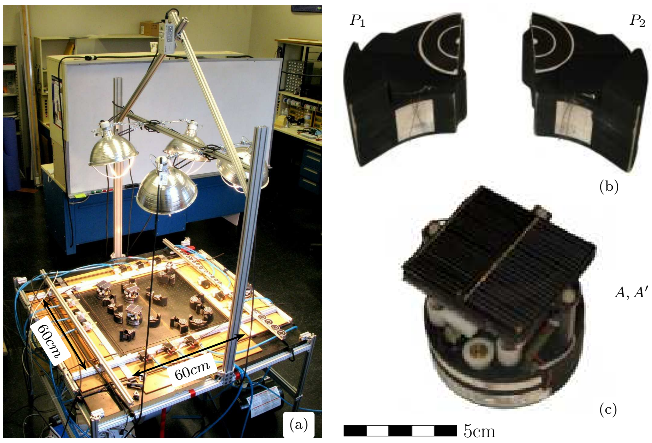

Fig. 2 Hardware of testbed. (a) Picture of the air-table showing the robots, the air-jets, the overhead lamps, and the overhead camera. (b) Picture of the two passive component types showing the tracking targets and complementary shapes. (c) The active robot showing solar cells, contact sensors, the spinning levers that pull bound passive heterodimers apart.

Images from the overhead camera are used to extract the number and position of *targets*, consisting of small, circular disks with a pattern of concentric light and dark rings, see Fig. 2(b). We detect targets in real time and use the data both in the feedback loop to exert control in Sec. 6 and open loop to estimate the system reaction rates and diffusion constants in Sec. 4.

The number of heterodimers is determined by selecting image processing parameters so that only whole targets register. Half of a target is attached each passive robot in such a way that when a heterodimer forms the two halves from a complete target, which is picked up by the vision system. The rotational symmetry of the targets simplifies the image processing by reducing the convolution of the target kernel from three to two dimensions, allowing sample rates of ≈ 1 Hz.

3 The Mathematical Model

This section describes *stochastic chemical kinetics* (SCK) [13] and the *chemical master equation* (CME), used to model the discrete number of each assembly type on the robotic testbed. The analogy with chemistry avoids having to model the configuration space of each robot, here SE(2), similar to the approach in [14]. When an input is added to the discrete model, the resulting system is called a *tunable reaction network*. This section also describes a *stochastic hybrid system* (SHS) model that extends SCK to include a continuous state variable needed to model the internal state of the controller of the closed loop system.

3.1 Stochastic Chemical Kinetics

The idea is to create a stochastic model that reflects our understanding of how chemical reactions occur at a microscopic level, as opposed to *mass action kinetics*, which is a deterministic model of the evolution of chemical concentrations. When the number of molecules involved in a set of chemical reactions grows, the approximations of mass action kinetics become very good. The large number of molecules averages stochastic effects away [15, Ch. 5.8]. However, when only a few molecules are involved, the stochastic nature of chemical reactions dominates the dynamics and requires explicit modeling.

Let S denote the set of chemical species. The *copy number* of each species is the number of instances of that particular species and is denoted by a capital N with the appropriate symbol as a subscript. The *state* $\mathbf{q}$ of the system is described by a vector of copy numbers and the set of all possible states is denoted by Q. Events that affect the state $\mathbf{q}$ are called *reactions*, which are indexed by a set L. If the state of the system is $\mathbf{q} \in Q$ before a reaction $l \in L$ and $\mathbf{q}' \in Q$ after the reaction, then we have

$$\mathbf{q}' = \mathbf{q} + \mathbf{a}_l,$$

where $\mathbf{a}_l$ is a vector that is specific to the reaction l. The chemical species that correspond to negative entries in $\mathbf{a}_l$ are called *reactants* and those that correspond to positive entries are called *products*. The *multiplicity* of a reaction $l \in L$ from a given state $\mathbf{q} \in Q$, denoted $M(\mathbf{a}_l, \mathbf{q})$, specifies the number of different ways the reactants of $\mathbf{a}_l$ can be chosen from state $\mathbf{q}$. In addition to the $\mathbf{a}_l$ vector, each reaction has associated with it a *rate constant* k_l, that depends on the underlying stochastic behavior of the interacting species.

In the robotic testbed the set of chemical species (robot types) is

$$S = \{A, P_{12}, A', P_1, P_2\}.$$

The symbol A stands for an active robot that is charged, A' is an uncharged active robot. The symbol P_1 and P_2 are the two different types of passive robots and P_{12} is a heterodimer of passive robots, see Fig. 1 and 2. The state of the robotic testbed is given by the vector of copy numbers

$$\mathbf{q} = (N_A, N_{P_{12}}, N_{A'}, N_{P_1}, N_{P_2})^T,$$

where, for example, the copy number of species A is denoted by N_A.

This paper considers the set of reactions in Fig. 1. For example, the reaction

$$P_1 + P_2 \xrightarrow{k_1} P_{12}$$

where two different passive robots form a dimer has the associated $\mathbf{a}$ vector

$$\mathbf{a}_1 = (0, 1, 0, -1, -1)^T.$$

Both P_1 and P_2 are reactants and P_{12} is a product. The multiplicity for this reaction is given by

$$M(\mathbf{a}_1, (N_A, N_{P_{12}}, N_{A'}, N_{P_1}, N_{P_2})^T) = N_{P_1} N_{P_2},$$

since there are N_{P_1} choices for the P_1 robot and N_{P_2} choices for the P_2 robot. Determining the rate constants for the system of robots is the topic of Sec. 4.2.

Stochastic Chemical Kinetics (SCK) defines a discrete state, continuous time *Markov process* with state space Q and the following transitions rates. The transition rate between $\mathbf{q}$ and $\mathbf{q}'$ is given by

$$k_l M(\mathbf{a}_l, \mathbf{q}), \qquad (1)$$

when $\mathbf{q}' = \mathbf{q} + \mathbf{a}_l$ and $\mathbf{a}_l$ is applicable in $\mathbf{q}$ (i.e. $\mathbf{q}'$ is non-negative). Given that the process is in state $\mathbf{q}$ at time t, the probability of transitioning to state $\mathbf{q}'$ within the next dt seconds is

$$k_l M(\mathbf{a}_l, \mathbf{q}) dt.$$

This property suffices to define the conditional transition probabilities of the stochastic process and together with an initial distribution over the states defines the Markov process that comprises the SCK model. This model is applicable to a set of interacting molecules if the system is *well mixed* [15,16]. In practice this assumption is difficult to verify. However, in our system of robots we can explicitly check the assumptions, since we can observe the position of all involved particles. A description of the procedures used to verify the well-mixed assumption is given in Sec. 4.1.

Conveniently, discrete state Markov Processes can be expressed via linear algebra in the following way. Fix an enumeration of Q and let $\mathbf{p}_i$ denote the probability of being in the ith state $\mathbf{q} \in Q$. The enumeration is arbitrary but assumed fixed for the remainder of this paper. The dynamics of the probability vector $\mathbf{p}$ are governed by the *infinitesimal generator* $\mathbf{A}$ defined as follows: All entries of $\mathbf{A}$ are zero except

- If $i \neq j$ and $\mathbf{q}_i + \mathbf{a}_l = \mathbf{q}_j$: $\mathbf{A}_{ij} = k_l M(\mathbf{a}_l, \mathbf{q}_i)$
- If $i = j$: $\mathbf{A}_{ii} = -\sum_m \mathbf{A}_{im}$.

By construction the rows of $\mathbf{A}$ sum to zero and all off-diagonal entries are non-negative. Probability mass functions over Q are expressed as row vectors and real valued functions $y : Q \to \mathbb{R}$ as column vectors. The dynamics of an arbitrary probability mass function $\mathbf{p}$ is governed by

$$\dot{\mathbf{p}} = \mathbf{p}\mathbf{A}, \tag{2}$$

the *Chemical Master Equation* (CME).

The convention of distinguishing between row vectors as probabilities and column vectors as functions highlights that functions and probabilities naturally operate on each other via the inner product. A probability distribution applied to a function produces an *expected value*. The inner product (denoted $\mathbf{py}$) is the expected value of a function $y : Q \to \mathbb{R}$ with probability mass function $\mathbf{p}$. When $\mathbf{p}$ is understood one we write $\mathbb{E}y$ instead.

The CME can be used to compute the change in expected value of an arbitrary function $y : Q \to \mathbb{R}$, since

$$\frac{d\mathbb{E}y}{dt} = \frac{d\mathbf{py}}{dt} = \frac{d\mathbf{p}}{dt}\mathbf{y} = \mathbf{pAy} = \mathbb{E}\mathbf{Ay}. \tag{3}$$

This equation gives *ordinary differential equations* (ODEs) for the expected value of scaler functions on the state space. In particular, they can be used to compute the statistical moments of random variables on Q. This fact and its extension to SHSs is used in Sec. 5 to prove stability of the proposed control scheme.

3.2 Tunable Reaction Networks

Since we are interested in controlling a chemical system we now turn our attention to inputs. The previous section described the ingredients of a reaction network: A list of involved species, a set of reactions, and rate constants. The first two parameters are not well suited as inputs since they require adding previously unmodeled elements to the structure of the network.

This paper treats inputs as adjustments to rate constants. There are several mechanisms that one might think of as affecting this change. For example, in biological system a change in rate could reflect the abundance of an enzyme that facilitates a reaction, a change in abundance of an inhibitor of a reaction, or some environmental parameter such as salinity or pH that has an effect on the efficiency of a reaction. In the case of the robotic example presented here, the rate change corresponds to changing intensities of the overhead lamps.

Whatever the mechanism, this type of input has some important features requiring consideration when used as an input for control. First, rate constants cannot be negative. This situation would correspond to backward reactions whose rates are depended only on the products but are independent of the reactants resulting in a non-causal reaction mechanism. Secondly, each of the aforementioned rate adjustment mechanisms typically saturates. For example, in the case an inhibitor based rate adjustment the reaction rate will never be higher than the uninhibited reaction. In the case of overhead lamps, they burn out when supplied with too much power.

Since the limits of saturation can be rescaled and absorbed into the rate constant we consider rate adjustments that modify some reaction so that (1) is instead given by

$$k_l u M(\mathbf{a}_l, \mathbf{q}), \tag{4}$$

where $u \in [0, 1]$. The resulting CME now has input u

$$\dot{\mathbf{p}} = \mathbf{p}\mathbf{A}(u), \tag{5}$$

where u is in the unit interval an modifies some off-diagonal terms of $\mathbf{A}$ linearly with the corresponding change in diagonal entries to conserve the zero row-sum property of $\mathbf{A}$.

3.3 Stochastic Hybrid System

Adding a continuous random variable whose dynamics depend on the discrete state $\mathbf{q}$ of a Markov process results in an SHS. This section is a brief description of the notation and some specific mathematical tools available for SHSs, for more information see [17–19].

3.3.1 Defining a Stochastic Hybrid System

The key features of a *Stochastic Hybrid System* (SHS) is that the dynamics of the system are stochastic and that the state are hybrid, meaning the state space of the system has the form $Q \times X$ where Q is some discrete set and $X \subseteq \mathbb{R}$ is continuous. The set of possible discrete states Q is typically finite or countably infinite. We use $z \in Z = Q \times X$ as shorthand for the pair $(\mathbf{q}, x)$. Let $\mathcal{Q}$, $\mathcal{X}$, and $\mathcal{Z}$ denote the stochastic processes on the various components of the state space. Note that each element of Q is a vector of copy numbers so we use a bold face $\mathbf{q}$ to denote the elements of Q.

In each discrete state, the dynamics of $\mathcal{X}$ are governed by a differential equation that can depend on both the continuous and discrete state,

$$\dot{x} = f(\mathbf{q}, x) \qquad f : Q \times X \to TX. \tag{6}$$

The dynamics of the discrete state $\mathcal{Q}$ are governed by a set of transitions, indexed by a finite set L. Each transition $l \in L$ has associated with it an intensity function

$$\lambda_l(\mathbf{q}, x) \qquad \lambda_l : Q \times X \to [0, \infty), \tag{7}$$

and a reset map

$$(\mathbf{q}, x) = \phi_l(\mathbf{q}^-, x^-) \qquad \phi_l : Q \times X \to Q \times X. \tag{8}$$

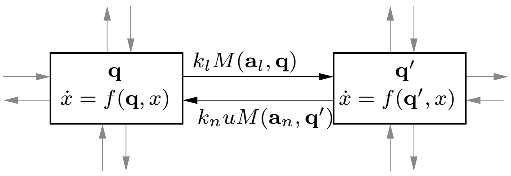
$\mathbf{q}$
$\dot{x} = f(\mathbf{q}, x)$
$k_l M(\mathbf{a}_l, \mathbf{q})$
$k_n u M(\mathbf{a}_n, \mathbf{q}')$
$\mathbf{q}'$
$\dot{x} = f(\mathbf{q}', x)$

Fig. 3 Two discrete states from a larger tunable reaction network. The grey arrows represent transitions to other states, which are not shown. The reaction $n \in L$ has in input $u \in [0, 1]$, otherwise the rates follow the dynamics from SCK. The state variables are related by $\mathbf{q} + \mathbf{a}_l = \mathbf{q}'$ and $\mathbf{a}_n = -\mathbf{a}_l$.

The intensity function is the instantaneous rate of the transition l occurring, so that

$$P(l \text{ occurs in } \ (t, t + dt) | \mathcal{Q} = \mathbf{q}, \mathcal{X} = x) = \lambda_l(\mathbf{q}, x)dt.$$

The reset map ϕ_l determines where the process jumps after a transition is triggered at $(\mathbf{q}^-, x^-)$ at time t. The minus in the superscript denotes the left hand limit of $\mathbf{q}$ and x at time t. We think of this limit as the state of the process immediately before the jump.

To model closed loop tunable reaction networks let the discrete states of the SHS be the set Q from the SCK description. The continuous part is used to model the controller state in Sec. 5. The resets maps and intensities of the SHS correspond to reactions and their associated rates. For a given reaction without input and index $l \in L$ the reset is given by

$$\phi_l(\mathbf{q}, x) = (\mathbf{q} + \mathbf{a}_l, x) \tag{9}$$

and the intensity is given by

$$\lambda_l(\mathbf{q}, x) = k_l M(\mathbf{a}_l, \mathbf{q}) \tag{10}$$

whenever $\mathbf{a}_l$ is applicable to $\mathbf{q}$. Similarly, for a reaction with input and index $n \in L$ the reset map is also given by

$$\phi_n(\mathbf{q}, x) = (\mathbf{q} + \mathbf{a}_n, x) \tag{11}$$

while the intensity is

$$\lambda_n(\mathbf{q}, x) = k_n u M(\mathbf{a}_n, \mathbf{q}) \tag{12}$$

whenever $\mathbf{a}_n$ is applicable to $\mathbf{q}$. In both cases the intensities are zero whenever a reaction is not applicable to a given state. The diagram in Fig. 3 represents the discrete states of such an SHS. The dynamics of the continuous state are only specified up to the function f and that the continuous state does not change during discrete transitions.

Note that our treatment differs from [18] which also uses SHS to model SCK. Here the discrete states are used to denote the copy numbers of chemical species as opposed to explicitly embedding them into the continuous state.

3.3.2 The Extended Generator

This section describes the *extended generator* $\mathcal{L}$ associated with an SHS. This operator is analogous to the *infinitesimal generator* of a discrete state Markov process described in Sec. 3.1. However, in the hybrid case the generator is a partial differential equation instead of a matrix. It allows the derivation of ODEs governing the dynamics of the statistical moments of scaler function on the state variables of an SHS.

Operator $\mathcal{L}$ in (13) is the extended generator for an SHS defined by (6)-(8). Let $\psi : Q \times X \to \mathbb{R}$ be a real valued *test function* on the states of an SHS and define

$$\mathcal{L}\psi(z) \equiv \frac{\partial \psi(z)}{\partial x} f(z) + \sum_{l \in L} (\psi(\phi_l(z)) - \psi(z)) \lambda_l(z). \tag{13}$$

The operator $\mathcal{L}$ relates the time derivative of the expected value of a test function ψ to $\mathcal{L}\psi$ via

$$\frac{d \ \mathbb{E}\psi}{dt} = \mathbb{E} \ \mathcal{L}\psi \tag{14}$$

[17]. Note the similarity to the infinitesimal generator in Sec. 3.1. When the test function ψ only depends on the discrete state

$$\psi(\mathbf{q}, x) = \psi(\mathbf{q}),$$

it can be written in vector form $\mathbf{y}$. In this case the operator $\mathcal{L}$ plays the same role as $\mathbf{A}$ in (3). This simplification when functions only depend on the discrete state is key in the stability proof of the feedback controller in Sec. 5.1.

The infinitesimal and extended generator for Markov processes are related, and in the discrete setting they are the same. However, in the continuous case, such as the continuous part of an SHS, the extended generator is defined for a larger class of test functions [19, Ch.1.4].

4 The Testbed Reaction Network

The reaction network description for our robotic testbed consists of four distinct reactions: two describe the spontaneous association and dissociation of passive robots P_1 and P_2, one describes the disassembly of P_{12} by active robots, and the last reaction describes recharging of active robots. Denote the rate constant for association and dissociation by the natural dynamics by k_1 and k_{-1}, for the disassembly reaction by k_2, and for the tunable recharge reaction by k_3. The rate constant for the tunable recharge reaction corresponds to the maximal physically possible rate, in this case highest operating intensity of the overhead lamps. These reactions are summarized in (15)-(17).

$$P_1 + P_2 \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} P_{12} \tag{15}$$

$$P_{12} + A \xrightarrow{k_2} P_1 + P_2 + A' \tag{16}$$

$$A' \xrightarrow{uk_3} A. \tag{17}$$

The discrete state space Q is finite and obeys the conservation equations

$$C_1 \equiv N_{P_1} + N_{P_{12}} = N_{P_2} + N_{P_{12}}, \tag{18}$$

$$C_2 \equiv N_A + N_{A'}, \tag{19}$$

where C_1 and C_2 are constants. The first relation (18) holds when the system has the same number of both types of passive robots (C_1 of each, which we ensure in our experiments), while (19) asserts that there are C_2 active robots that can either be in a charged or discharged state. As a consequence of (18) and (19), N_{P_1}, N_{P_2}, and A' can be expressed in terms of N_A, $N_{P_{12}}$, and the constants C_1 and C_2. Instead of five different species we can keep track of only two. For the remainder of this paper we will assume that

$$\mathbf{q} = \begin{pmatrix} N_A \\ N_{P_{12}} \end{pmatrix} \in \mathbb{N}^2$$

and note that the copy number for the missing species can be reconstructed from this reduced state.

4.1 Checking the Well-Mixed Condition

There are several equivalent definitions of what it means for a system to be well-mixed. Basically, all definitions are sufficient conditions for guaranteeing that a process is Markov and that each *possible* combination of reactants for a particular reaction $\mathbf{a}_l$ is equally likely to be involved in the next reaction. While being well-mixed in this sense is a strong assumption, it allows for the characterization of a reaction by a single parameter, the rate constant k_l. For the remainder of this section we use the definition of well-mixedness from [16]. For alternative conditions see [15, Ch. 7.2]. The two conditions that must be checked are that: (a) the reactants are uniformly distributed throughout the environment and (b) that the reactants diffuse through the reaction domain faster than they react.

To estimate the distribution of the different types of robots on the air-table we decomposed it into a 11×11 grid and extracted the occupancy statistics for each grid box from video data. Fig. 4 shows the resulting distributions. The red area in the center of each plot is roughly at the same level and indicates a uniform distribution. The area of low occupancy around the perimeter results from the fact the position of each robot is estimated at its center yet geometric constraints keep the center away from from the air-table border.

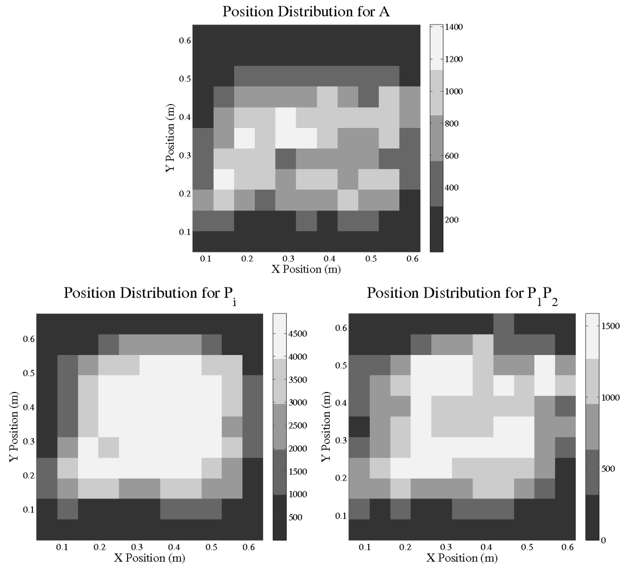

Fig. 4 Observed distribution of robots on air-table. The top figure corresponds to active robots, A or A', the lower left plot correspond to passive robots P_1/P_2, and the lower right figure to heterodimers P_{12}. These plots demonstrate that the occupancy of parts on the air-table is roughly uniform on the table. The area of low occupancy around the perimeter is due to the geometry of the components interacting with the boundary of the air-table.

The diffusion coefficient for a robot is defined as

$$D = \frac{\mathbb{E}\ r^2(t)}{4t},$$

where $r(t)$ denotes the random displacement of the robot as a function of time. We used the targets described in Sec. 2 to track the position of different robot types. We averaged over multiple experiments as well as the instances of each robot type to compute the expected value. The resulting estimates for the diffusion coefficient are given in Tab. 1. The subscripts of D indicates what robot type the diffusion coefficient was calculated for. For example, $D_{P_{12}}$ is the diffusion coefficient of heterodimers.

Combined with the rate constants measured in Sec. 4.2 we conclude that condition (a) and (b) are approximately met. The testbed is well-mixed and the SCK model is appropriate.

4.2 Characterizing Rate Constants

One method to determine rate constants is to measure the average waiting time between reactions from a known state. This quantity, together with the known inverse relationship between the reaction rate and average waiting time, yields an estimate of the rate [6].

Although useful in simulation, one drawback of this method is that one needs to repeatedly re-initialize the system to gather statistical data, which is tedious and time consuming. An exception is k_3, which was measured in this way. The reason is that the recharge reaction represents a change in internal state, which is easy to re-initialize.

For the other rate constants we take a different approach. We average multiple longer trajectories all *starting* from the same initial condition. However, the system is allowed to continue evolving for a set amount of time, possibly undergoing many reactions. This has the advantage that each re-initialization gives much more information than a single waiting time. We then fit this empirical average to solutions of the CME (2).

We determined the remaining rate constants k_1, k_{-1} and k_2 in two steps. First, we gathered trajectories starting from $N_{P_{12}} = N_A = 0$ with $u = 0$ (lights off). This way the disassembly reaction and recharge reaction do not influence the natural dynamics. We then used `MATLAB` to numerically fit the CME solution with the two free parameters k_1 and k_{-1} to the empirical average, minimizing the mean squared error, see Fig. 5(a).

Using the values previously determined for k_3, k_1, and k_{-1} we then used the same approach (this time with $u = 1$) for determining the only remaining parameter in the CME solution, k_2. The resulting curve fit is shown in Fig. 5(b).

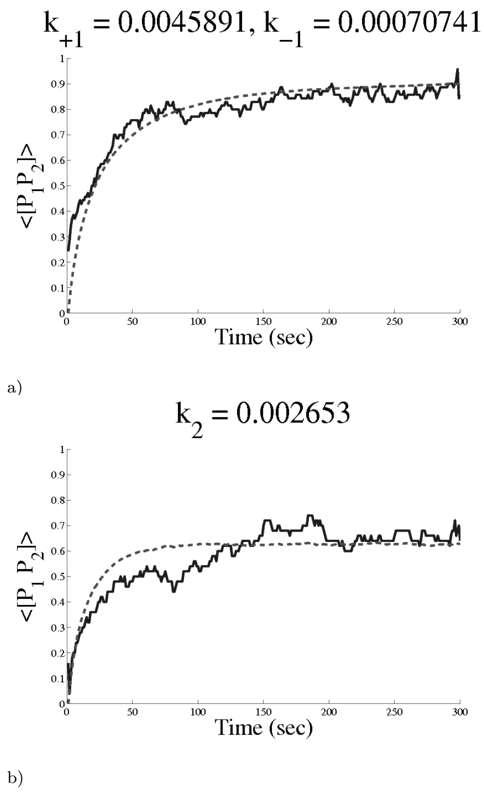

Fig. 5 Curve fitting results used to determine rate constants.

Parameter	Estimate	Uncertainty	Units
k_1	0.0046		$\frac{reaction}{sec\ number^2}$
k_{-1}	0.00071		$\frac{reaction}{sec\ number}$
k_2	0.0027		$\frac{reaction}{sec\ number^2}$
k_3	0.08		$\frac{reaction}{sec\ number}$
$D_A/D_{A'}$	0.0018	0.0002	$\frac{m^2 sec}{sec}$
D_{P_1}/D_{P_2}	0.0015	0.0001	$\frac{m^2 sec}{sec}$
$D_{P_{12}}$	0.00083	0.00001	$\frac{m^2 sec}{sec}$

Table 1 Estimates of rate constants and diffusion coefficients.

5 Integral Control for Tunable Reaction Networks

This section describes an integral feedback controller for tunable reaction networks introduced in Sec. 3.2. The idea for this control scheme is simple: The integral of the instantaneous error is fed back to the input of the tunable reaction network in a direction that counteracts the cumulative error.

Difficulties arise both from the fact that the input saturates and that the stochastic nature of a tunable reaction networks will randomly move the output function both closer and further away from any given setpoint. This situation is in stark contrast with the well developed theory of integral feedback for LTI systems.

The inherent stochasticity can be addressed by controlling the expected value of the output function instead of trying the force the system to stay in a particular set of states. While this is a much weaker form of control, it is also a much more feasible given the types of inputs that are typically available in systems that are well modeled by SCK.

When controlling the expected value of the output instead of directly controlling the output of the system, one has to worry about the variance. We do not present theoretical results concerning the variance around the controlled mean in general. However, we do present estimates of the variance for the robotic testbed in Sec 6.

Because the input saturates, we need to ensure that the setpoint is actually achievable with the limited control effort that is available. This restriction results in a *controllable region* of setpoints for which the proposed controller works correctly.

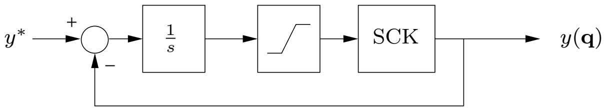

Fig. 6 Block diagram of the proposed integral feedback controller for tunable reaction networks. The block labeled SCK with an input, together with the saturation block is a block diagram description of a tunable reaction network. The other blocks form a simple integral feedback controller.

5.1 Integral Control

The discrete state $\mathbf{q}$ of a tunable reaction network develops according to (5). Suppose there is some output of the system that we want to regulate. Let $\mathbf{y}$ be the a vector corresponding to an output function $y : Q \to \mathbb{R}$. A typical example is $y(\mathbf{q}) = N_{S_i}$, the copy number of species S_i.

The desired setpoint of the output is denoted by y^* and the cumulative error from the setpoint by x. The dynamics of x are given by

$$\dot{x} \;=\; \gamma(y(\mathbf{q}) - y^*), \tag{20}$$

where γ is an integrator gain. This equation is the last remaining parameter to define an SHS together with the reset maps and intensities (9)-(12). This construction gives a class of SHS that has an input $u \in [0,1]$ it inherits from the dynamics of the discrete states.

In order to model the closed loop system, a way to express saturation of the input is needed. Let $h : \mathbb{R} \to \mathbb{R}$ be given by

$$h(x) = \begin{cases} 0,\ x \leq 0 \\ x,\ 0 < x \leq 1 \\ 1,\ 1 < x. \end{cases}$$

The dynamics of the closed loop integral feedback controller are given by setting

$$u = h(x). \tag{21}$$

A block diagram of the control system is shown in Fig. 6. The remainder of this section is dedicated to analyzing the closed loop system.

Our approach to showing that this controller works is to look at the test function

$$\psi(\mathbf{q}, x) = x.$$

Since the reset maps (9) and (11) do not modify the continuous component, the sum in (13) evaluates to zero, resulting in

$$\frac{d\mathbb{E}x}{dt} = \mathbb{E}\gamma(y(\mathbf{q}) - y^*)) = \gamma(\mathbb{E}y(\mathbf{q}) - y^*). \tag{22}$$

If the closed loop system is stochastically stable, in the sense that probability distributions of states approaches a fixed invariant distribution, then by (22)

$$\mathbb{E}y = y^*. \tag{23}$$

The controller works in expected value when the system is in steady state. Now, the problem of showing correctness of the controller reduces to showing that the system is stochastically stable or *ergodic*, i.e. that the system always approaches a unique steady state distribution.

5.2 Ergodicity

We use a Lyapunov function argument [20, THM 5.1] to show that the closed loop SHS is ergodic. This allows us to set the LHS in (34) to zero and argue that the controller works in steady state. We show that the system is ergodic for some reference values y^* and give sufficient conditions for ergodicity for a range of y^*.

Denote the infinitesimal generator matrices of minimum and maximum input by $\mathbf{A}_m = \mathbf{A}(0)$, $\mathbf{A}_M = \mathbf{A}(1)$ and the corresponding steady state probability mass functions by $\mathbf{p}_m$ and $\mathbf{p}_M$ respectively.

Theorem: Let $\mathbf{A}(u)$ be the generator of a tunable reaction network and $\mathbf{y}$ the vector corresponding to an output function $y : Q \to \mathbb{R}$ of the discrete state. The feedback controller proposed in (21) results in a closed loop system with a stationary distribution that has $\mathbb{E}y = y^*$ when y^* is in the *controllable region*, $\mathbf{p}_M\mathbf{y} < y^* < \mathbf{p}_m\mathbf{y}$.

Note: If $\mathbf{p}_M\mathbf{y} > \mathbf{p}_m\mathbf{y}$, then the theorem applies with the sign in (20) and the upper and lower limits of the controllable region reversed.

Proof: Let $\mathcal{Z}$ be the SHS corresponding to the closed loop system. By [20, THM 5.1], $\mathcal{Z}$ is ergodic when there exists a function $V : Z \to \mathbb{R}^+$ with the property that $V(z) \to \infty$ as $|z| \to \infty$ and

$$\mathcal{L}V(z) \leq -f(z) \quad \forall z \notin C \tag{24}$$

for some compact region C and positive function f. [1]

For our system, we define the function $\widehat{V}$ to be

$$\widehat{V}(\mathbf{q}, x) = \begin{cases} \;\;x + c^+(\mathbf{q}) & \text{for } x > 0 \\ -x + c^-(\mathbf{q}) & \text{for } x < 0, \end{cases}$$

where c^+ and c^- depend on $\mathbf{q}$. Note that the function $\widehat{V}$ is neither differentiable (required to apply $\mathcal{L}$) nor positive (required by theorem) since the offsets can be negative. To address this problem, let V be a function that agrees with $\widehat{V}$ when x is outside some interval $[v_{min}, v_{max}]$ for all $\mathbf{q} \in Q$, and is both non-negative and twice differentiable. This function always exists since Q is finite and $\widehat{V}$ increases with $|x|$ so that $\widehat{V}$ is positive for sufficiently large $|x|$.

[1] The theorem has some technical preconditions, which are fulfilled in our case, namely that all compact sets are petite see [20]. This follows from [21, THM 4.1], [19, THM 27.6] and the fact that every Feller process is also a T-process.

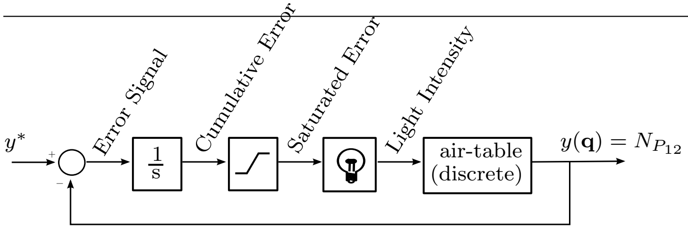

Fig. 7 Block diagram of the proposed control system. Only the air-table state and output signal are discrete, all other signals are continuous.

Let the compact region required by the theorem be $C = Q \times [\min(v_{min}, 0), \max(v_{max}, 1)]$. Since we are only interested in V outside C, we look at the cases when the feedback input is saturated at either $u = 0$ or $u = 1$. This situation simplifies the analysis, since the transition intensities $\lambda(\mathbf{q}, x)$ are independent of x in the saturated regions. We now argue that for some range of setpoints y^* we can find c^+ and c^- to make V a Lyapunov function in the sense of (24).

Choosing $f = \epsilon$ and considering saturation at $u = 1$ first, we rewrite the conditions of (24) in vector from,

$$\mathbf{y} - y^*\mathbf{1} + \mathbf{A}_M\mathbf{c}^+ \leq -\epsilon\mathbf{1}. \tag{25}$$

Let $\tilde{\epsilon}$ be an arbitrary vector with strictly positive entries, then (25) can be rewritten as

$$\mathbf{y} - y^*\mathbf{1} + \mathbf{A}_M\mathbf{c}^+ = -\tilde{\epsilon}. \tag{26}$$

We want to determine when this equation has a solution for $\mathbf{c}^+$. Note that

$$\mathbf{A}_M\mathbf{c}^+ = -\tilde{\epsilon} + y^*\mathbf{1} - \mathbf{y}$$

has a solution only if $(-\tilde{\epsilon} + y^*\mathbf{1} - \mathbf{y})$ is in the column space of $\mathbf{A}_M$, which we write $(-\tilde{\epsilon} + y^*\mathbf{1} - \mathbf{y}) \in \mathrm{Col}\mathbf{A}_M$. Equivalently

$$(\mathrm{Col}\mathbf{A}_M)^\perp \perp (-\tilde{\epsilon} + y^*\mathbf{1} - \mathbf{y}) \tag{27}$$
$$(\mathrm{Nul}\mathbf{A}_M^T) \perp (-\tilde{\epsilon} + y^*\mathbf{1} - \mathbf{y}) \tag{28}$$
$$(\mathbf{p}_M)^T \perp (-\tilde{\epsilon} + y^*\mathbf{1} - \mathbf{y}) \tag{29}$$
$$0 = \mathbf{p}_M(-\tilde{\epsilon} + y^*\mathbf{1} - \mathbf{y}) \tag{30}$$
$$0 = -\mathbf{p}_M\tilde{\epsilon} + y^* - \mathbf{p}_M\mathbf{y}, \tag{31}$$

where Nul denotes the right null space and a superscript $\perp$ the orthogonal complement. Because $\tilde{\epsilon}$ has arbitrary, strictly positive entries and entries and $\mathbf{p}_M$ has non-negative entries by definition a solution for $\mathbf{c}^+$ exists when

$$\mathbf{p}_M\mathbf{y} < y^*.$$

Similarly, for saturation with $u = 0$ we get

$$\mathbf{p}_m\mathbf{y} > y^*.$$

Thus the system is ergodic if

$$\mathbf{p}_M\mathbf{y} < y^* < \mathbf{p}_m\mathbf{y}. \tag{32}$$

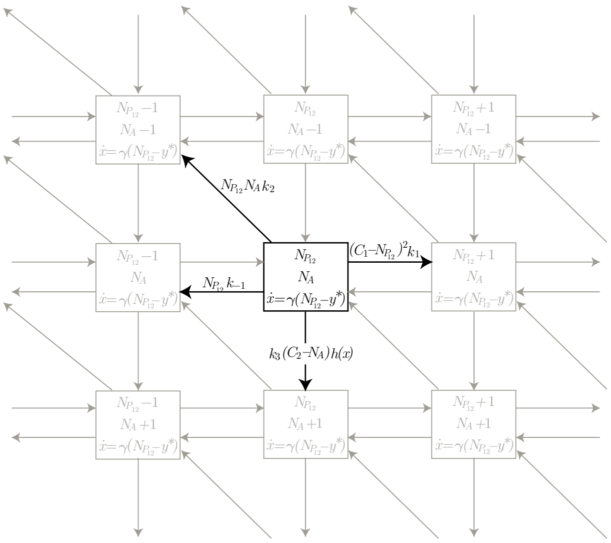
$N_{P_{12}}-1$ N_A-1 $\dot{x}=\gamma(N_{P_{12}}-y^*)$
$N_{P_{12}}$ N_A-1 $\dot{x}=\gamma(N_{P_{12}}-y^*)$
$N_{P_{12}}+1$ N_A-1 $\dot{x}=\gamma(N_{P_{12}}-y^*)$
$N_{P_{12}}N_A k_2$
$N_{P_{12}}-1$ N_A $\dot{x}=\gamma(N_{P_{12}}-y^*)$
$N_{P_{12}}k_{-1}$
$N_{P_{12}}$ N_A $\dot{x}=\gamma(N_{P_{12}}-y^*)$
$(C_1-N_{P_{12}})^2k_1$
$N_{P_{12}}+1$ N_A $\dot{x}=\gamma(N_{P_{12}}-y^*)$
$k_3(C_2-N_A)h(x)$
$N_{P_{12}}-1$ N_A+1 $\dot{x}=\gamma(N_{P_{12}}-y^*)$
$N_{P_{12}}$ N_A+1 $\dot{x}=\gamma(N_{P_{12}}-y^*)$
$N_{P_{12}}+1$ N_A+1 $\dot{x}=\gamma(N_{P_{12}}-y^*)$

Fig. 8 A schematic representation of the closed loop SHS. The boxes represent discrete states and the arrows represent transitions. Each box shows both the discrete state it represents and the ODE describing the continuous states. An arbitrary state $(N_A, N_{P_{12}})^T$ is highlighted in black. The transition intensities for all transitions leaving $(N_A, N_{P_{12}})^T$ are shown next to the arrows.

Furthermore, by (22) the expected value of y tracks the reference value y^* when it is in the controllable region.

■

The proof does not rely on any special structure of $\mathbf{A}(u)$ nor the value of γ. Therefore, the theorem is generally applicable to tunable reaction networks with saturating inputs, which is a good model for a large class of systems. For example, adding more types of passive species and active robot types would not change the applicability of this theorem. However, if the input changes a reaction rate that does not have much effect on the output, then the controllable region is small. As a result, while the controller is applicable it is only useful with an effective input. Since y is an arbitrary output function this scheme could be used to control the expected value of other quantities, as long as they depend only on the discrete state.

Also, as a side effect of our approach to the proof we have shown that the system is ergodic, which means that the steady state ensemble average and long term time average are the same. This feature is important since the above theorem concerns the ensemble average at different points in time. However, from an application perspective the time average of individual trajectories is often more important.

6 Integral Control Applied to the Robotic Testbed

This section combines the mathematical model of the robotic testbed developed in Sec. 4 with the results from the previous section to solve the control problem stated in Sec. 1.With the mathematical model for the testbed in place we can state the problem more formally. *Control the system such that* $\mathbb{E}N_{P_{12}} = y^*$ *by adjusting the intensity of the overhead lamps.*

To apply the theorem from from Sec. 5 let the output function $y : Q \to \mathbb{R}$ be given by

$$y(\mathbf{q}) = N_{P_{12}}.$$

The resulting feedback controller is shown in Fig. 7. The closed loop SHS is characterized by the state transition diagram shown in Fig. 8 and the extended generator of the closed loop SHS

$$\begin{aligned}
&\mathcal{L}\ \psi(N_{P_{12}}, N_A, x) \qquad (33)\\
&= \frac{\partial \psi(N_{P_{12}}, N_A, x)}{\partial x}\gamma(N_{P_{12}} - y^*)\\
&+ (\psi(N_{P_{12}} + 1, N_A, x) - \psi(N_{P_{12}}, N_A, x))k_1(C_1 - N_{P_{12}})^2\\
&+ (\psi(N_{P_{12}} - 1, N_A, x) - \psi(N_{P_{12}}, N_A, x))k_{-1}N_{P_{12}}\\
&+ (\psi(N_{P_{12}} - 1, N_A - 1, x) - \psi(N_{P_{12}}, N_A, x))k_2 N_{P_{12}} N_A\\
&+ (\psi(N_{P_{12}}, N_A + 1, x) - \psi(N_{P_{12}}, N_A, x))x(C_2 - N_A).
\end{aligned}$$

As argued in Sec. 5 (22)-(23) setting $\psi = x$ gives

$$\frac{d\ \mathbb{E}x}{dt} = \mathbb{E}\ \gamma(N_{P_{12}} - y^*), \qquad (34)$$

resulting in correct behavior when the process is ergodic.

6.1 Experimental Results

We implemented the proposed controller on the robotic testbed described in Sec. 2. The generator matrix $\mathbf{A}(u)$ is defined by (15)-(17) with the parameters found in Sec. 4. In the following experiments the number of P_1 and P_2 pairs is $C_1 = 10$ and the number of active robots is $C_2 = 4$.

To show the controller's the tracking capability we tracked two periods of a square wave. The low and high setpoints were 0.7 and 0.8 (corresponding to 7 and 8 P_{12}). Both of the setpoints are inside the empirically determined controllable region for this system, 0.60-0.86.

To determine the upper and lower limits of the controllable region, the system was operated in open loop with both $u = 0$ and $u = 1$ and allowed to equilibrate. After reaching equilibrium the expected value of the output function $N_{P_{12}}$ was estimated, which by (32) are the upper and lower limits of the controllable region.

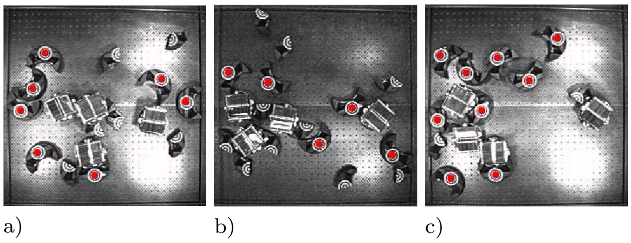

Fig. 9 Frames taken from the overhead camera feed. Each crescent shaped dimer P_{12} detected by the vision system is labeled with a red dot (color online). The four square objects in each frame are the solar panels of the active robots. The number of red dots in each frame corresponds to the output function $N_{P_{12}}$ of the state. Note that different intensities of the overhead lamps, corresponding to different inputs, can seen in the reflections on the air-table. The input in is lowest in b) and highest in c). Frames b) and c) are chosen to show that the system output can temporarily be outside the controllable region due to random fluctuations. The output function in b) is smaller than 6.0 and larger than 0.86 in c).

Intuitively, the size of the controllable region depends on the influence the global input has on the tunable reaction network. Even if the active robots were to recharge instantly (unlimited input), there would still be a non-zero equilibrium distribution of $N_{P_{12}}$, depending on the relative rates in (15) and (16). The size of the region depends on the network structure.

In this particular example, if there were more P_1 and P_2 robot pairs but the same number of active robots the controllable region would be smaller. If there were more active robots and the same number of passive P_1 and P_2 robots the size of the controllable region would be larger. The input only changes the recharge rate of active robots, if there are more of them it has a larger effect on the steady state behavior.

Snapshots from the overhead camera during the experimental trials are shown in Fig. 9. The combined results of 25 trajectories are shown in Fig. 10. We let each trajectory run with a setpoint of 0.7 for 5 minutes (a half period) before recording data, which allowed transients resulting from the manual initialization to dissipate. After the warm up period we collected 20 minutes of data for each trajectory.

This experiment demonstrates the controller tracking a reference signal in mean (Fig. 10(b)(c)). This experiment also demonstrates the fundamental stochasticity in the system. The spread in Fig. 10(b)(c) is not due to measurement uncertainty or noise, but a fundamental property of the stochastic system we are controlling.

Next, we present simulation experiments exploring how the variance of the copy number relates to the integrator gain γ in the example system. The proof for tracking in mean did not depend on the value of γ,

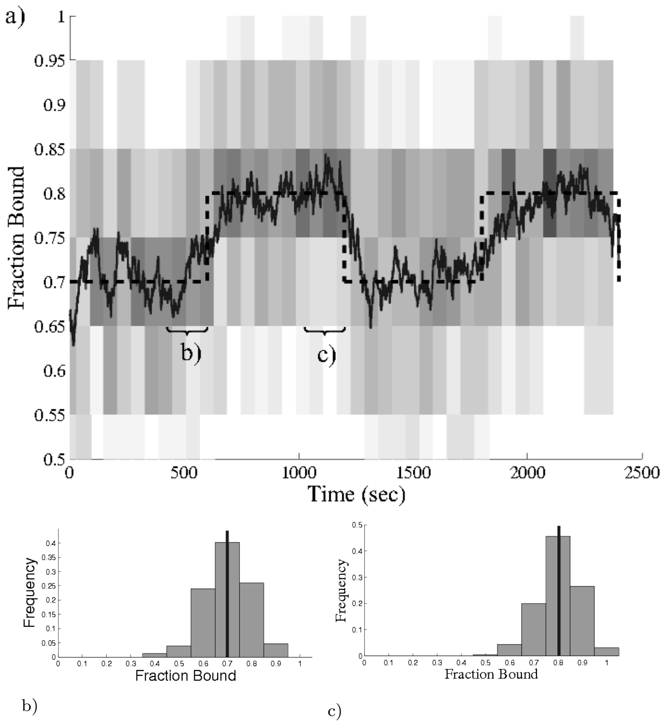

Fig. 10 Tracking data from robotic testbed. This is the average of 25 different experiments. Each experiment has 10 P_1 and P_2 robots, and 4 active robots. The grey shading in the background corresponds to the fraction of trajectories with that number of pairs. Darker regions correspond to higher occupancy. The two histograms on the bottom show the fraction of possible dimers taken for the last half of each step, b) c). The vertical line is the computed mean, which demonstrates correct controller behavior.

so the proposed controller will always yield the desired mean steady-state copy number as long as the setpoint is in the controllable region. However, it might differ in the degree of fluctuation around the correct mean.

6.2 Variance of the Controlled System

To investigate the dependence of the closed loop system variance on the integrator gain and setpoint we turn to simulations.

While SSA [16] is an efficient way to simulate trajectories for SCK, the non-constant transition intensities in the controlled system make this method difficult to apply. Instead, we use a stochastic integration algorithm with the parameters from Sec. 4 to simulate closed loop trajectories to obtain an estimate of the variance. The open loop variance is computed exactly, by first numerically finding the input u such that the associated infinitesimal generator has the correct steady state

$$\mathrm{Nul}(\mathbf{A}(u)^T)^T\mathbf{y} = y^*,$$

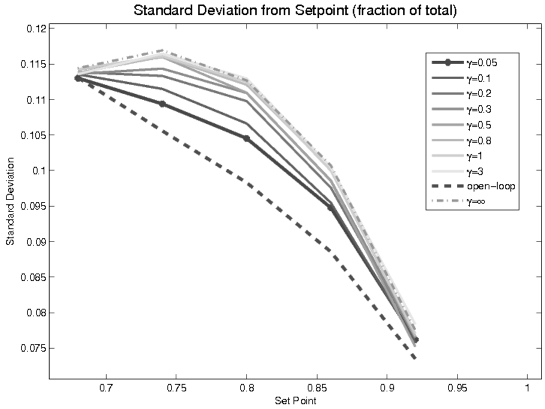

Fig. 11 Standard deviation of output for different setpoints and integrator constants γ. The dashed line at the bottom corresponds the standard deviation when the system is under open loop control. The dash-dot line at the top corresponds to the standard deviation of the system when driven with a bang-bang controller and the input is always saturated. Each standard deviation is estimated from 200 simulation trajectories.

and then using this steady state to compute the variance

$$\mathbb{E}(y - y^*)^2.$$

The relation between setpoint, integrator gain, and standard deviation at steady state are shown in Fig. 11. Each data point was collected by setting γ and estimating the standard deviation at steady state. This approach limits smallest value of γ we can feasibly simulate, since small values slow down the system dynamics and make simulations prohibitively computationally expensive.

We observe that less aggressive values of γ result in a smaller standard deviation of the output. The upper and lower limits of the standard deviation correspond to open loop and bang-bang control. Another interesting feature of Fig. 11 is that the standard deviation of the output is less sensitive to γ if the reference y^* is close to the edge of the controllable region.

The dependence of the variance on the integrator gain suggests that very low values of γ result in the lowest steady state variance. However, there is a tradeoff between speed of convergence and steady state variance. When γ is small, then the dynamics of the integrator variable x are slow. When the integrator dynamics are much slower than the open loop dynamics then the small integrator gain is the limiting factor in reaching steady state. In many cases slow response times are undesirable, even if these result in a smaller steady state variance. Lowering the integrator timescale by increasing γ can decrease the convergence time at the cost of a higher steady state variance.

Note, that the controllable region in the simulated system is shifted up from the physical system. This is likely due to the charging behavior of the active robots, which is modeled as exponential in the simulation, but actually has a different distribution. Each disassembly takes roughly the same amount of energy. However, regardless of the exact location of the controllable region, we expect the qualitative behavior of the variance to be similar, especially since the network structure is the same. What is important about the controllable region is that setpoints inside it can be reached.

7 Conclusions and Future Work

We proposed an integral feedback controller for controlling the average copy number of an arbitrary species in a system modeled as a tunable reaction network. We prove that the controller tracks a reference in mean and demonstrate the approach on an robotic experimental platform. We also present some preliminary simulation results regarding the variance of the copy number as a function of the integrator gain and setpoint. We are currently working on analytical results describing the steady state variance of the control scheme.

Finally, we would like to emphasize the generality of our approach. This control scheme works for any tunable reaction network and requires no tweaking of the integrator gain γ as long as the reference is in the controllable region, which is easy to measure experimentally. Also, the tunable reaction network model is quite general since the variable rate input can model a wide variety of physical mechanisms. While we presented an example system with relatively large components, we believe that the presented or similar approaches will be especially useful in the context of many microscopic components.

In the future we would like to find ways to decentralize the controller by using local estimates of the global output. In particular, we want to implement this control scheme with other chemical reactions. For example, the continuous state could correspond to a high copy number species where the mass action kinetics approximation works well. Such an implementation would open the door to using the proposed control scheme in many other settings, such as biological systems.

Acknowledgements

We would like to thank Sheldon Rucker, Michael McCourt, and Yi-Wei Li for their dedication and hard work in designing and building the robots. We would also like to thank Alexandre Mesquita and João Hespanha for their helpful discussions about proving ergodicity.

References

1. E. Saeedi, S. Kim, H. Ho, and B. A. Parviz, "Self-assembled single-digit micro-display on plastic," in *MEMS/MOEMS Components and Their Applications V. Special Focus Topics: Transducers at the Micro-Nano Interface*, vol. 6885, pp. 688509–1 – 688509–12, SPIE, 2008.
2. P. W. K. Rothemund, "Folding dna to create nanoscale shapes and patterns," *Nature*, vol. 440, pp. 297–302, Mar. 2006.
3. M. Boncheva, D. A. Bruzewicz, and G. M. Whitesides, "Millimeter-scale self-assembly and its applications," *Pure Appl. Chem*, vol. 75, pp. 621–630, 2003.
4. H. Onoe, K. Matsumoto, and I. Shimoyama, "Three-dimensional micro-self-assembly using hydrophobic interaction controlled by self-assembled monolayers," *Microelectromechanical Systems, Journal of*, vol. 13, pp. 603–611, Aug. 2004.
5. Massimo, Mastrangeli, S. Abbasi, agdas Varel, C. van Hoof, C. Celis, and K. F. Bhringer, "Self-assembly from milli to nanoscales: Methods and applications," *IOP Journal of Micromechanics and Microengineering*, vol. 19, no. 8, p. 37pp, 2009.
6. S. Burden, N. Napp, and E. Klavins, "The statistical dynamics of programmed robotic self-assembly," in *Conference Proceedings ICRA 06*, pp. 1469–76, 2006.
7. P. J. White, K. Kopanski, and H. Lipson, "Stochastic self-reconfigurable cellular robotics," *IEEE International Conference on Robotics and Automation (ICRA04)*, pp. 2888–2893, 2004.
8. H. El-Samad, J. P. Goff, and M. Khammash, "Calcium homeostasis and parturient hypocalcemia: An integral feedback perspective," *Journal of Theoretical Biology*, vol. 214, pp. 17–29, Jan. 2002.
9. M. Yim, W. Shen, B. Salemi, D. Rus, M. Moll, H. Lipson, E. Klavins, and G. Chirikjian, "Modular self-reconfigurable robot systems," *IEEE Robotics Automation Magazine*, vol. 14, pp. 43 –52, march 2007.
10. P. White, V. Zykov, J. Bongard, and H. Lipson, "Three dimensional stochastic reconfiguration of modular robots," in *Proceedings of Robotics Science and Systems*, (Cambridge MA), June 2005.
11. N. Ayanian, P. White, Á. Hálász, M. Yim, and V. Kumar, "Stochastic control for self-assembly of XBots," in *ASME Mechanisms and Robotics Conference*, (New York), August 2008.
12. B. Donald, C. Levey, I. Paprotny, and D. Rus, "Simultaneous control of multiple mems microrobots," in *The Eighth International Workshop on the Algorithmic Foundations of Robotics (WAFR)*, December 2008.
13. D. A. McQuarrie, "Stochastic approach to chemical kinetics," *Journal of Applied Probability*, vol. 4, pp. 413–478, Dec 1967.
14. K. Hosokawa, I. Shimoyama, and H. Miura., "Dynamics of self assembling systems: Analogy with chemical kinetics," *Artificial Life*, vol. 1, no. 4, pp. 413–427, 1994.
15. N. V. Kampen, *Stochastic Processes in Physics and Chemistry*. Elsevier, 3rd ed., 2007.
16. D. T. Gillespie, "Exact stochastic simulation of coupled chemical reactions," *Journal of Physical Chemistry*, vol. 81, no. 25, pp. 2340–2361, 1977.

17. J. P. Hespanha, “Modeling and analysis of stochastic hybrid systems,” *IEE Proc — Control Theory & Applications,* Special Issue on Hybrid Systems, vol. 153, no. 5, pp. 520–535, 2007.
18. J. P. Hespanha and J. Abhyudai, “Stochastic models for chemically reacting systems using polynomial hybrid systems,” *International Journal of Robust and Nonlinar Control*, vol. 15, no. 15, pp. 669 – 689, 2005.
19. M. Davis, *Markov Processes and Optimization.* Chapman & Hall, 1993.
20. S. P. Meyn and R. L. Tweedie, “Stability of Markovian processes III: Foster-lyapunov criteria for continuous-time processes,” *Advances in Applied Probability*, vol. 25, no. 3, pp. 518–548, 1993.
21. S. P. Meyn and R. L. Tweedie, “Stability of Markovian processes II: Continuous-time processes and sampled chains,” *Advances in Applied Probability*, vol. 25, no. 3, pp. 487–517, 1993.