Markov decision process based time-varying optimal control for colloidal self-assembly

Xun Tang∗ Michael A. Bevan∗∗ Martha A. Grover∗∗∗

∗ Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: xtang38@gatech.edu).
∗∗ Johns Hopkins University, Baltimore, MD 21218 (e-mail: mabevan@jhu.edu).
∗∗∗ Georgia Institute of Technology, Atlanta, GA 30332 (e-mail: martha.grover@chbe.gatech.edu).

Abstract: Crystals made of periodically well-ordered nano- and/or micro-scale elements can interact with light to give novel properties. These perfect crystals have applications in a wide range of areas. For example, invisibility cloaks that reroute light transmission make objects disappear. However, manufacturing such perfect crystals still remains challenging. Here, we propose a low-dimensional Markov decision process based dynamic programming framework to optimally control a colloidal self-assembly process for perfect crystal fabrication. Based on the simulation results, we demonstrate that an open-loop control policy identified with the proposed framework is able to reduce the defective assemblies from 46% of uncontrolled to 8% of controlled production. Moreover, when feedback is available, a closed-loop optimal finite-horizon control policy can further reduce the defective assemblies down to 5% out of 100 independent simulation runs.

Keywords: optimal control, feedback control, open-loop control, dynamic programming, stochastic control.

1. INTRODUCTION

Crystals composed of periodically well-ordered small elements possess the ability to interact with light at specific wavelengths. Micro-scale particles suspended in solution are called colloids, and assemblies made of colloids can be ordered at the same length scale as light wavelength. A recent study by Ni reported the feasibility of making an ultrathin skin cloak out of nanoantennas that is able to reroute light and render objects invisible (Ni et al. (2015)). Other applications include: adaptive optics (Holtz and Asher (1997)), reconfigurable circuit elements (Yang et al. (2009)), semiconductors (Velev and Lenhoff (2000)) and so on. Despite the attractive applications, manufacturing perfectly ordered colloidal assemblies over larger scales is still challenging. Currently available manufacturing methods can be generally defined into two categories: top-down and bottom-up methods (Biswas et al. (2012)). Top-down fabrication such as lithography, is achieved largely by patterning features. It starts from larger dimensions and reduces to the required values (Gates et al. (2005)). On the other hand, bottom-up fabrication, like self-assembly, builds up assemblies from smaller components (Ariga et al. (2008)).

Top-down approach has shown success as reported in Biswas et al. (2012) and its references. However, bottom-up approach provides a more promising fabrication for large quantities with less waste and better quality con-

* This work is sponsored by National Science Foundation, through collaborative grant Grant #1124678 (Grover) and #1124648 (Bevan) through the Cyber-Enabled Discovery and Innovation program.
ics simulation was used to generate samples for construction of a Markov state model (Bowman et al. (2014)). The Markov state model was then used to calculate the optimal control policy (i.e. input voltage trajectory) using dynamic programming, via a Markov decision process framework (Puterman (2005)). With the objective to maximize the global crystallinity $\psi_6$, 98 out of 100 independent experiments (i.e. a 98% yield) produced a perfect assembly (i.e. assembly with a higher than 0.96 $\psi_6$ value) with the control policy compared to a 60% yield under a constant high input (Tang et al. (2016)).

The closed-loop policy calculated in the previous work was time-independent, here we calculate a time-dependent closed-loop optimal policy, using a finite-time formulation of the Markov decision process. The objective is now to maximize global crystallinity at a pre-specified terminal time. The incorporation of time-dependence brings in more flexibility by yielding different control policies at different updating time point. It also adds physical insight into the optimality of switching between a high and low input level (Swan et al. (2014)). In fact, in our past work with a time-independent policy, we also observed similar switching patterns, due to the system transitioning between different regions of the state space. In addition to closed-loop policies, the Markov state model are also used to calculate an optimal open-loop policy to maximize expected final crystallinity. This further allows us to understand the optimality of simple toggling schemes.

2. COLLOIDAL ASSEMBLY SYSTEM

2.1 Experimental Setup

In this study, we focused on a SiO$_2$ colloidal particle assembly process (Tang et al. (2016), Juárez et al. (2012)). In this system, about 300 SiO$_2$ particles with a radius of 1.5 $\mu$m are suspended in deionized water in a container made of glass microscope cover slips (50 mm x 24 mm x 150 $\mu$m). Four separate, tunable 1 MHz AC electrode tips are attached to the edge of the container to generate a non-homogeneous electric field. The system is monitored in real time using a microscope.

In an AC electric field mediated colloidal self-assembly system, colloidal particles become induced dipoles when the field is turned on. The interaction between the electric field and the particles, together with the particle-particle interactions, provides the driving force for the crystallization. A detailed study of the electric-field mediated assembly mechanism is presented by Juárez and Bevan (2009). In order to make a direct comparison to the toggling scheme, only two input levels are considered here: $\lambda = 0.2$ and $\lambda = 19.7$, where $\lambda$ is a dimensionless representation of voltage, indicating the strength of the driving force (Tang et al. (2016)).

Fig. 1 shows the experimental setup (Fig. 1a), top-view zoomed in to the assembly system (Fig. 1b), with electrode distance as 96 $\mu$m, particle configurations under the low input $\lambda = 0.2$ (Fig. 1c), and the high input $\lambda = 19.7$ (Fig. 1d,e,f). The local hexagonal order $C_6$ is colored in blue, and the global hexagonal order $\psi_6$ colored in red. Index 6 associated with the order parameters indicates the maximum number of neighboring particles of each particle in a 2-dimensional plane. The system exhibits a fluid state at $\lambda = 0.2$ while reaching a perfect crystal at $\lambda = 19.7$ (Fig. 1f). However, due to the strong attractive force, grain boundaries and defective structures also tend to form under the high input (Fig. 1d,e). In this study, we focus on removing the grain boundaries from the 2-dimensional assemblies.

2.2 Brownian Dynamics (BD) Simulation and Order Parameters

Building an optimal control policy for a stochastic process requires a thorough understanding of the system dynamics. With an accurate model, it is then possible to effectively study the dynamics over a wide range of repetitions and under extreme conditions. A Brownian dynamics simulation was built and validated against experiments to simulate the system dynamics (Edwards et al. (2013)). The Cartesian coordinates of each particle, together with its velocity, are used as the input for the simulation; therefore it is a 4N-dimensional simulation model, where N is the number of particles. The BD simulation provides an accurate prediction of the system by tracking the individual particle positions. The detailed background on the Brownian dynamics simulation is given in Edwards et al. (2014).

However, keeping track of all the particle coordinates makes the simulation a time-consuming process. More importantly, using the entire Cartesian coordinates as a state description hinders the calculation of a control policy due to the high dimensionality. Instead, it is often useful to calculate a reduced set of coordinates, which captures the overall system features of interest. This would introduce errors to the reduced-order model, but it makes the state classification feasible and the control approach tractable. The dimensionality reduction can be accomplished using physical intuition, computer algorithms, or a combination
of both. A detailed review on the techniques and progress can be found in Rohrdanz et al. (2013), van der Maaten et al. (2009). Based on physical understanding, we identified two order parameters, \( C_6 \) and \( \psi_6 \), to describe the system evolution by distinguishing the crystalline state (high \( \psi_6 \) and high \( C_6 \)), the fluid state (low \( \psi_6 \) and low \( C_6 \)), and defective states (low \( \psi_6 \) but high \( C_6 \)). Specifically, \( C_6 \in (0,6) \) quantifies the local order of the system, and \( \psi_6 \in (0,1) \) describes the global order of the assembly.

2.3 Markov State Model and Optimal Control Policy

A Markov state model (MSM) is characterized by a set of discrete states \( S \), here defined by \( (\psi_6, C_6) \), and a probability transition matrix \( P(a) \) for each input \( a \), i.e., \( \lambda \), \( P(a) \) is composed of transition probability \( P(a)_{ij} \), which denotes the probability of the system to be in state \( j \), after a transition time of \( \Delta t \), given the current state as \( i \), under input \( a \) (Bowman et al. (2014)).

The accuracy of an MSM depends on state discretization and the number of samples. The choice of \( \Delta t \) is also important and non-trivial; a small value is desired to capture more dynamics, and to update the control more frequently for better performance. However, due to possible unmodeled phenomena using the reduced order parameters, a larger \( \Delta t \) is needed for model accuracy. Therefore a balance between the two is desired (Prinz et al. (2011)). After trial-and-error with a range of different \( \Delta t \) values, this study is primarily focused on a control update time of \( \Delta t = 100 \) s.

Sampling plays an important role in model accuracy, and here we used a dynamic sampling method to generate samples from the Brownian dynamics model. The input is switched randomly among levels of \( \lambda = 0.2 \) and \( \lambda = 19.7 \) at 100 s intervals, which is the same as the transition time \( \Delta t \). The design of a dynamic input trajectory aimed to find states excited by switches. Simulations were initiated and repeated (to account for stochastic effects) in different discrete states to ensure enough samples for a common state space under each of the two input levels. To enrich the sampling, samples from constant input trajectories were also included.

With the simulation samples available, the state space was then discretized. The discretization should be fine enough to distinguish configurations that lead to different dynamics, but too fine a discretization would lead to sampling issues in model building and computational issues in control policy calculation. As a balance, \( \psi_6 \) was discretized into 50 evenly spaced intervals, and \( C_6 \) was discretized into 120 evenly spaced intervals; therefore, a total of 6000 discrete states were defined. With \( \Delta t \) value, discrete state space, and the simulation samples defined, we estimated the transition matrix \( P(a) \) using a moving window strategy elaborated as Counting Method 1 in Prinz et al. (2011).

After the MSMs were built, a finite-horizon Markov decision process based optimization problem was solved. A finite Markov decision process is defined by \( T, S, A, P(a) \), where \( S, A, P(a) \) are defined previously, and \( T \) is the collection of the discrete time epoch \( k \). If \( T \) is finite, the process is called a finite-horizon MDP, otherwise, an infinite-horizon MDP. In the finite-horizon MDP considered here, the optimization is achieved over a finite number of steps, with the objective \( J_a \) defined as:

\[
J_a(x) = E \{ R(x_{t_f}, a_{t_f}) \} 
\]

\[
J^*(x) = \sup_{a \in A} J_a(x) \tag{2}
\]

where \( E \) is the expectation operator, \( a \in A \) is the control action, \( x \in S \) is the discrete system state, and \( t_f \) is the final time instant. \( R(x, a) : S \times A \rightarrow R \) is the one-stage reward function obtained when the system is in state \( x \) and control action \( a \) is taken, and is defined as \( R(x, a) = \psi^2 \). The design of the finite-horizon MDP aimed to maximize the final time crystallinity. The optimal value function and the optimal policy \( a^* \in A \) are defined as

\[
a^*(x) = \arg\{ \sup_{a \in A} J_a(x) \} = \arg\max_{a \in A} J^*(x) \tag{3}
\]

The optimal control policy was solved with dynamic programming via a backwards conduction algorithm embedded in the MATLAB MDP Toolbox (Cros (2009)). An infinite-horizon control policy was also defined and calculated in the same way as in Tang et al. (2014) as a comparison. Order parameter \( C_6 \) was not explicitly included in the objective for reasons that: first, a high \( \psi_6 \) state automatically ensures a high \( C_6 \) value, due to their physical meanings; second, \( C_6 \) contributes more towards state classification than the control policy according to our previous findings. Based on past work (Tang et al. (2016)), 98% of yield was achieved within 1000 s; here we focus our investigation on a process time of 900 s.

The key steps of our approach are summarized in Fig. 2: first, we conducted dimensionality reduction with initial samples from high-dimensional simulations; then, we built Markov state models in the reduced state space with additional samples; after that, we computed and evaluated the optimal control policy; finally, based on the control performance, the procedure is repeated with more samples for improvement. The computational challenge here resides in building the accurate MSMs, which requires a large enough sampling to include important dynamics for each input. To solve the optimal control policy, it only takes several seconds on a 3.40 GHz Intel(R) Xeon(R) CPU with 16.0 GB memory.
The MSMs were also used to identify the optimal open-loop control policy for a 900 s process with a 100 s updating time as follows: 1. we enumerated all the possible input profiles that end with a high input for the last 100 s; 2. we used the MSMs to predict and identify the optimal policy as the one that gave the highest final stage crystallinity quantified by $E[\psi_6^2]$, based on Markov Chain Monte Carlo simulations.

3. RESULTS

Fig. 3 shows the 900 s finite-horizon optimal control policy in the reduced state space, with each table presenting the control policy for the corresponding 100 s time interval. The control action is found by looking up the corresponding color at the interception of the two order parameters, and is updated every 100 s. Since the high input level is used for the last 100 s (800 s to 900 s) to maintain the structure, for simplicity it is omitted here. Note that, the local order parameter $C_6$ is normalized over its theoretical maximum value $\delta$ in the plotting.

In open-loop toggling schemes, odd (100 s) intervals use $\lambda = 19.7$, and even intervals use $\lambda = 0.2$ on a time basis. However, the optimal control policy features the use of $\lambda = 0.2$ in the upper region of the order parameter coordinates, which corresponds to defective states; and uses $\lambda = 19.7$ in the fluid states. The alternation between the high and low input is based more on the state instead of time. Moreover, the control policy at 600 s and 700 s intervals shows a reducing use of the low input, this also mimics the time-based open loop toggling scheme. These observations indicate that the optimal control policy incorporates both time and system state as updating factors.

To evaluate the performance, BD simulations were conducted using three sets of control schemes: 1. toggling: 60sTog, 100sTog, 180sTog, and 300sTog, using the open-loop toggling scheme starting with $\lambda = 19.7$, and alternating with $\lambda = 0.2$ every 60 s, 100 s, 180 s, or 300 s for a 900 s process; 2. optimal open-loop control policy that updates every 100 s: 100sOL; 3. closed-loop feedback optimal control policy that updates every 100 s: InfCL and FinCL, developed from the infinite-horizon and the 900 s finite-horizon framework. All the BD simulations were initiated from fluid states with 10 different configurations, and a total of 100 independent simulation runs were included to account for the stochastic effects.

Fig. 4a summarizes the $\psi_6$ evolution of the four toggling schemes. Due to the periodic use of a high and low input, the system assembled and relaxed along the process. Switching times of 60 s and 180 s gave a similar final $\psi_6$ after 900 s, while 100 s switching time gave the highest final $\psi_6$ and the 300 s switching scheme the lowest. This indicates that switch to slowly leads to unnecessary relaxation which acts as re-initiation (300 s switching); on the other hand, switch too fast results in insufficient relaxation for defects correction.

The optimal open-loop policy (i.e. 100sOL) gave the same final $\psi_6$ value as the 100 s toggling scheme after 900 s, but with less use of low input levels indicated by the relaxations in the trajectories in Fig. 4b. Indeed, MSM results showed that the 100 s toggling scheme is a near-optimal open-loop control policy with a final reward of $\psi_6^2 = 0.967$, while the optimal open-loop policy is $\psi_6^2 = 0.973$. This similar performance is resulted from the similarities between the two policies: 100sOL mimics the toggling scheme except for a successive use of the high input at the first and the last 200 s. The BD results further revealed several possibilities: 1. the inaccuracy in the MSMs leads to a non-optimal open-loop policy for the BD simulation; 2. multiple near optimal open-loop policies exist for the BD system; 3. even if the MSMs are not perfect, we are still able to provide good-performing open-loop policies.

Fig. 4c summarizes the optimal open-loop and closed-loop controlled (InfCL, FinCL) results with respect to $\psi_6$. In general, the feedback control gave a slightly higher final
ψ_6. The final reward in BD simulation was ψ_6^2 = 0.925 for the closed-loop infinite-horizon policy, ψ_6 = 0.936 for the closed-loop finite-horizon policy, ψ_6 = 0.897 for the 100 s toggling scheme, and ψ_6 = 0.893 for the optimal open-loop policy (100sOL). Different from the open-loop control, feedback control used low input levels less frequently, indicated by fewer relaxations in the trajectory. This indicates relaxation is not always necessary. The finite-horizon policy produced a better final result than the infinite-horizon policy. This is because the time-dependence gave more flexibility in the policy design, but one should also notice that the infinite-horizon policy was not designed for a definite stopping time point.

The comparison of the four best performing policies in terms of yield is presented in Fig. 4d. Overall, the closed-loop finite-horizon control policy gave the highest yield. Although at a 95% significance level, the 100 realization simulations did not show a significant difference statistically, we could still see the trend that: 1. feedback has the potential to improve the performance compared to open-loop controls; 2. potential improvement could be achieved by adding more flexibility (time-dependence) to the policy.

A closer investigation on several individual simulations reveals more information on the mechanism. For simplicity, only the 900 s closed-loop finite-horizon control policy (FinCL) and the 100 s toggling scheme (100sTog) are presented. Fig. 5a shows a case where the system achieved a perfect crystalline state under both cases. With the optimal control policy, λ = 0.2 was used to relax the system only when defects were formed. After one relaxation-reassembly step, the system reached the perfect state at an early stage in both cases. High input was then used in the control policy to maintain the structure; while the toggling scheme was still alternating with low input. This indicates that although toggling worked in some cases, relaxation was not always necessary. Fig. 5b shows a case where a perfect assembly was achieved with the optimal control policy, but not with the toggling scheme. Instead of toggling, a successive use of low input fully resolved the defects and allowed the system to achieve a perfect crystal after 300 s under the control policy. These observations indicate that feedback not only provides a stopping criterion, it could also shorten the process.

### Table 1. Yield and assembly time in BD

<table>
<thead>
<tr>
<th></th>
<th>Assembly Time (s)</th>
<th>LB Time (s)</th>
<th>Yield</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uncontrolled</td>
<td>167</td>
<td>504</td>
<td>54%</td>
</tr>
<tr>
<td>Tog60s</td>
<td>284</td>
<td>370</td>
<td>86%</td>
</tr>
<tr>
<td>Tog100s</td>
<td>310</td>
<td>357</td>
<td>92%</td>
</tr>
<tr>
<td>Tog180s</td>
<td>330</td>
<td>444</td>
<td>80%</td>
</tr>
<tr>
<td>Tog300s</td>
<td>439</td>
<td>660</td>
<td>52%</td>
</tr>
<tr>
<td>100sOptOL</td>
<td>306</td>
<td>354</td>
<td>92%</td>
</tr>
<tr>
<td>Inf100sCL</td>
<td>284</td>
<td>339</td>
<td>91%</td>
</tr>
<tr>
<td>Fin100sCL</td>
<td>299</td>
<td>329</td>
<td>95%</td>
</tr>
</tbody>
</table>

Fig. 5c shows a case where the toggling and the optimal control policy both failed. Several reasons could contribute to this: 1. a longer assembly time was needed; 2. the inaccuracy in MSMs leads to a non-optimal policy for the BD simulation; 3. more input levels were needed. Fig. 5d shows the averaged optimal input (FinCL) from the BD simulation as a comparison to the toggling scheme (100sTog). The rationale behind the optimal input trajectory is as follows: at early stages, low input was used...
frequently to heal the defects, while high input was used more later on to maintain the assembly.

The final analysis is the time needed to produce a perfect crystal. Results are summarized in Table 1: 1. the average assembly time (Assembly Time), defined as the averaged first time a perfect crystal was achieved; 2. the lower bound assembly time (LB time) for the system to produce a perfect assembly every time, defined by taking 900 s as the least amount of time needed for the failed simulation trials; 3. the final yield of perfect assembly out of the 100 BD simulations.

The uncontrolled case, i.e. applying $\lambda = 19.7$ throughout the process, required the least amount of time among the assembled ones. This is because if the system did not get into a defective state, it forms a perfect crystal rapidly under the strong attractive force. However, the high input introduced defects which were hard to eliminate, therefore resulting in the lowest yield of 54% in the end. Similarly, the toggling scheme switched every 300 s required a short amount of assembly time but with a low yield. The toggling scheme switched every 60 s gave a shorter assembly time but a higher yield compared to its 180 s counterpart. This means that switching frequently could improve both the yield and shorten the assembly time. However, switching too rapidly would instead hurt the overall yield therefore elongating the comprehensive assembly time, see the 100 s switching results. In all the cases, the closed-loop finite-horizon optimal control policy required the least amount of assembly time overall, and it also gave the highest yield after 900 s. To deal with the failed 5% runs, the process can be extended until satisfying results are achieved.

4. CONCLUSIONS

In this paper, we applied a Markov decision process framework on a stochastic colloidal self-assembly process. We demonstrated that the defective assemblies can be reduced from 46% of uncontrolled to 5% of controlled simulations, with an average assembly time reduced by 30%. We also demonstrated that with the proposed framework, an optimal open-loop policy could be calculated, and a 92% yield of perfect crystal could be achieved. We further conclude that further improvements can be achieved by introducing a time-dependence to the control policy, given the time-varying and the time-independent closed-loop policies. The framework can be generalized to any system where global actuators exist, dynamics models are available to predict system response to the actuators, and feedback is available to close the control loop. Future work includes controlling larger system size with different particle shapes, and improving MSM accuracy. Experimental implementation is under way.

REFERENCES


