

**Modeling and Control of State-Affine
Probabilistic Systems for Atomic-Scale Dynamics**

Grant number: FA9550-04-1-0183

Martha A. Gallivan
School of Chemical & Biomolecular Engineering
Georgia Institute of Technology

ABSTRACT

Under this research grant, the framework and tools were developed for model reduction of atomic-scale many body systems. The state-affine mathematical structure of the original and reduced-order models enabled the implementation of control through dynamic programming and estimation through an extended Kalman filter. In the framework developed here, the state of the high-dimensional stochastic system is first quantified using a high-dimensional pair correlation function. This state is then reduced using linear and nonlinear principal component analysis and is discretized using self-organizing maps. To create the dynamic model, a cell map is constructed using short simulations to quantify input-dependent transitions between the discrete states. The error associated with the model reduction was quantified and analyzed, and a method for predicting this error was proposed.

Specific applications in materials processing were considered, which motivated and guided the development of the model reduction framework and tools. A model of gallium arsenide deposition was used to demonstrate the model reduction framework. A second modeling study in the molecular architecture of hyperbranched polymers was performed, and enabled a comparison of common themes and system specific features between the two different applications.

I. OBJECTIVES AND SIGNIFICANCE

The goal of this research program was to develop a systematic control methodology for state-affine probabilistic systems. This class of systems represents local interactions among a large number of discrete particles, and is particularly important for dynamics at the nanometer scale, where the interactions among many atoms, molecules, or spins require a probabilistic description. Stochastic realizations may be performed using Monte Carlo simulations, but it is difficult to use them directly in standard algorithms for dynamic optimization and controller design. The three main objectives of this research program, as articulated in the original proposal, were

1. to develop the theory underlying a model reduction idea for kinetic Monte Carlo simulations

2. to develop systematic tools for the reduction process and for control of the reduced models
3. to investigate the applicability of the theory and procedures in several application areas of interest for communications, computation, and materials development.

Dynamic models based on physical principles, such as conservation of mass, momentum, and energy, are routinely used in the design of systems and controllers. However, models of atomic scale phenomena, which often are high dimensional, nonlinear, and stochastic, are not typically used in design. Air Force systems demand new high performance materials and electronic devices, and due to the myriad possible design and processing options for nanostructured materials, models are needed to design, optimize, and control the processes. One illustrative example is the heterostructure field effect transistor under study at the Air Force Research Laboratory at Wright Patterson Air Force Base. Professor Gallivan spent the summer of 2006 at

AFRL working with a team there to model the degradation dynamics of these new electronic devices for high-speed high-power radar applications. Although these nanostructured transistors initially perform well, they degrade over time. It is believed that the degradation and failure modes depend strongly on the processing conditions and initial material defects. Due to the small scales of the device, the models require a consideration of a small number of point defects at random locations in the device.

Two additional applications were considered in this AFOSR-sponsored project. A lattice Monte Carlo simulation of gallium arsenide surface processing was used to motivate, guide, and demonstrate the model reduction framework developed here. Gallium arsenide is a III-V compound semiconductor used in high-speed transistors, and its interface properties must be tightly controlled in layered electronic devices such as transistors. Hyperbranched polymers are the second application considered in this program. Their unique molecular architecture makes them promising materials for sensors and catalysis, although the relationship between the process inputs and the final molecular structure has not previously been well understood.

The modeling framework proposed and developed here has wide potential applicability to a range of materials processing applications that are modeled by stochastic many body simulations. The framework and tools draw on a range of approaches in the model reduction and data mining disciplines. The model building approach requires significant computation, but it can be efficiently parallelized, and the resulting reduced-order model requires minimal computation, which is critical for real-time controller implementation. The model building process is largely automated by the tools described here, although some user input is required to select an appropriate pair correlation function to define the original state space. In this report,

the modeling framework and tools are first presented, followed by a demonstration of the modeling and control strategy, after which the applications are discussed.

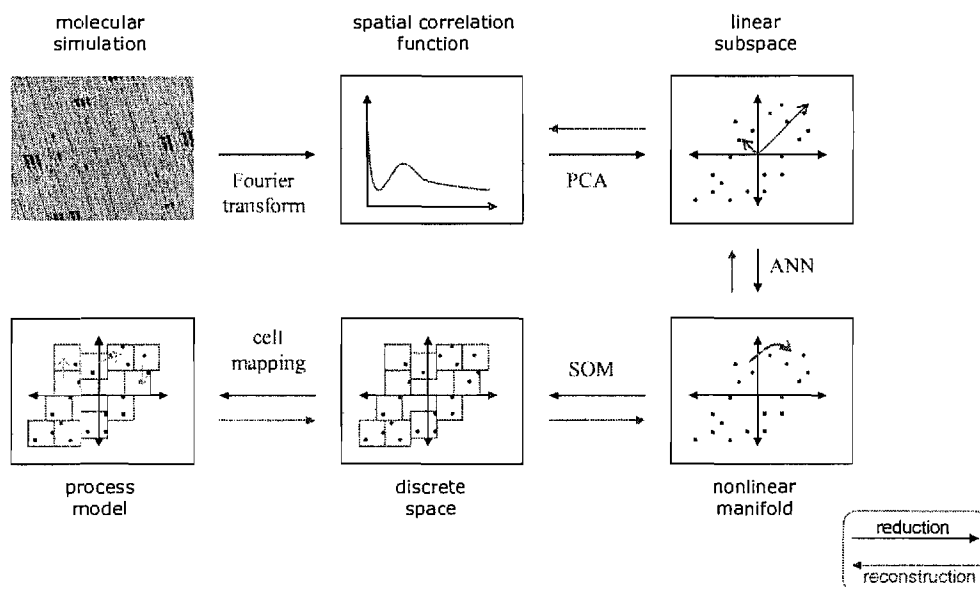


Figure 1. Reduced-order modeling framework and tools for atomic simulations.

II. SIGNIFICANT ADVANCEMENTS

The concept behind the model reduction framework is illustrated by Figure 1. The labels on the arrows of Figure 1 give examples of tools that can be used to implement each of the steps. Some of the concepts behind this reduction approach were previously proposed and used by Gallivan and Murray [1]. A goal of the work supported by this grant was to propose a complete reduction process, and to identify and demonstrate tools that could be used to automate the reduction.

1. Quantification of the state space

A first critical step in constructing a dynamic model is to define the state space of the system, which is a non-trivial task for an atomic-scale many body simulation. Although the state of the stochastic simulation can be defined as the position of each atom in the system, each realization of the stochastic simulation will be expected to yield a completely different state. However, the control objectives for the system are based on overall statistical measures that define material structure and properties, and this is not expected to change significantly between realizations that are performed under the same nominal process inputs.

One way to think about this issue is to consider symmetries in the stochastic simulations. Periodic boundary conditions are used, so translational symmetries exist. Thus, it is not the exact position of each atom that is relevant to the state, but the relative positions of the atoms. Furthermore, most of the atoms in the simulations are indistinguishable: in the GaAs simulations, there are two atom types: gallium and arsenic; in the polymer simulations, there are two types of molecules, which we refer to as A_2 and B_3 . However, in the entire simulation domain there are thousands or millions of each atom. Thus, the state, especially in a reduced-order model, need not contain the exact position of each atom, but instead should include only the *relative* positions of each atom *type*. This concept is already used in analyzing atomic scale simulations, through the pair correlation function. It is still a high-dimensional description, but captures the frequency of pairs of atoms as a function of their distance and/or angle. In surface simulations, such as in the GaAs system, a height-height correlation function is often used to quantify the simulation results, and is a pair correlation function based on the height of the surface at different distances in x- and y-directions that define the surface. This height-height

correlation function can in fact be computed using the Fourier transform of the surface height, so one interpretation of a pair correlation function is in analogy to taking a Fourier transform of the simulated surface, for the purpose of removing symmetries. In our GaAs demonstration, we instead use a pair correlation function based on the distance between *steps* in the surface, which is instead more like the Fourier transform of the *derivative* of the surface height. Due to the small changes in surface height due to atomic-height steps, we found that the height-height correlation function was extremely sensitive to noise in the simulations, while the step-step correlation function better captured the discrete changes due to atomic height steps using a small number of modes. Thus, the selection of the best pair correlation function for a particular system cannot be fully automated at this time, and requires selection based on an understanding of the key material structures in the system. Future work could involve a more automated selection process, although it should also be expected that some understanding of the physical system will aid in the modeling and reduction process.

3. Tools for model reduction

A significant accomplishment of this grant is the identification of a set of tools that can automate the model reduction process in a computationally tractable manner. These tools include principal component analysis (PCA), nonlinear principal component analysis (NLPCA), self-organizing map (SOM), cell mapping, and k-nearest neighbors interpolation.

Once the pair correlation function has been identified for the atomic simulation, methods are required to reduce its dimension in order to construct a reduced-order model. The dimension of the pair correlation function depends on the simulation size, and should be expected to scale with either the length or volume of the simulation domain, depending on whether the correlation function includes distances only, or also directions. However, beyond a certain correlation

length, the correlation functions should reach a plateau value (this is a requirement in the atomic simulations to ensure that the simulation domain is sufficiently large). Additional redundant features often exist in these correlation functions, which can be associated with dominant material structures. As a result, the next step in the modeling framework is to search for lower-order behavior in the molecular simulations, using the method of snapshots commonly used for model reduction of partial differential equations, such as in fluid flow [2]. In this approach, simulations are run under a wide range of inputs and initial conditions, and principal component analysis (also called proper orthogonal decomposition) is used to identify a lower order state space spanned by orthogonal basis vectors. The coordinates in this new basis are then used as the coordinates in a new reduced-order model. *Nonlinear* correlations between the states may also exist, and we have demonstrated the use of nonlinear principal component analysis, which is most efficiently performed after linear correlations have been removed using principal component analysis via autoassociative neural nets (Gallivan in 2004, Proceedings of *DYCOPS*).

A further step we take in our modeling method is to discretize this low order state, which enables construction of a Markov chain model based on input-dependent transitions. We have accomplished this discretization using self-organizing maps (SOM) [3], although other methods for clustering data could alternatively be used. The self-organizing map converges with minimal computation required, and has the additional beneficial feature of creating a projection of the nonlinear space onto a two-dimensional plane (by organizing the clusters), which can aid in visualization of trajectories of the system. Our overall approach to state reduction was published by Oguz and Gallivan in 2005 in the *International Journal of Nonlinear and Robust Control*.

Of course, it is also possible to not discretize the state space, and instead to construct a low-order dynamic model on the continuous space after performing the principal component analysis. This

is accomplished in applications such as fluid flow using a Galerkin projection on the original system of partial differential equations. However, in the atomic simulations, there is no PDE for the projection, and it is not clear what should be the mathematical form of reduced-order dynamic model. One option is to project onto the probabilistic master equation that defines the probability distribution of the entire system over all possible realizations on the stochastic simulations. However, due to the extremely large or even infinite number of possible configurations of the system, direct mathematical operation on this master equation is prohibitive.

For this reason, we have chosen to discretize the reduced-order state space and construct a dynamic model based on input-dependent transitions between the discrete states in the model. As with any reduced-order model based on the method of snapshots, it is not expected that this model will be able to predict states that are significantly different from those in the original set of snapshots; however this is not a particular limitation of the discrete state space, but rather of the method of snapshots. The physical interpretation of each discrete state is that it represents a typical type of surface or material configuration that was observed during the method of snapshots training. Similar snapshots are grouped by the SOM into equivalence classes, so that each snapshot is associated with one of the discrete states. Conversely, each discrete state is associated with a point in the reduced continuous state space, and therefore is also associated with the pair correlation function that can be reconstructed from its point in the reduced continuous state space.

Once the discrete state space has been defined, the dynamic model is constructed by running short simulations for a particular predefined interval (typically this interval would be based on time). A simulation, or a set of realizations, is run beginning with a configuration in each

discrete state, and is run for the discretized time step under a particular process input. Here we discretize the inputs into discrete levels, and construct transitions for each of these levels (Interpolation between these levels is discussed in the next section). At the end of each simulation, the pair correlation function and corresponding PCA reduced coordinates are computed, and are matched to the discretized state (or states) that best matches, based on a Euclidean distance of the PCA coordinates. By performing short simulations beginning in each discrete state, and at each input level, a dynamic model is constructed. The computational requirements of this reduction depend on the number of discrete states, the number of input levels, and the number of realizations performed for the purpose of averaging out random noise. In our GaAs demonstration, we use approximately 150 states, 8 input levels, and 10 realizations of each run. Since each realization takes approximately 1 hour, the computational time on a single processor is significant but not intractable. However, the realizations can be distributed across multiple processors, which thus reduces the time by a factor equal to the number of processors used.

3. Comparison of mathematical structures for the dynamic model

The reduced-order model as described in the previous section is fully discretized: in time, in the state, and in the inputs. This model can then be used to predict the evolution of the pair correlation function for time-varying but piecewise constant process inputs. However, one can also think about interpolating between these discrete points, which potentially could reduce the error due to the discretization.

A first method for interpolation was explored using the framework of cell maps. A simple cell map is a discretized version of a continuous-state dynamic model, in which each discretized state is mapped to exactly one other state at a future time (under a specified input). This is the case in

our reduced-order model. However generalized cell maps can be constructed in which a single state maps to multiple states with differing probabilities [4]. We explored this approach as a method to interpolate between discrete states, but due to the coarse discretization of our state space, spreading due to numerical diffusion was large, causing the overall prediction error to increase in comparison to simple cell mapping. Generalized cell mapping could provide an improvement under a finer discretization of the state space, but due to the significant computation associated with running each short atomic-scale simulation, we recommend simple cell mapping over generalized cell mapping given the current computational technology.

The problem with the generalized cell map is that the number of states with non-zero probability grows large as the number of time steps advances. To avoid this problem, while still interpolating, we investigated the k -nearest neighbors interpolation method on the discrete space. In particular, we focused on $k = 2$, so that the two best matching units of the current state of the reduced-order model are identified. The probabilities of each of these two states are computed based on the Euclidean distance between the current state and two best matching states. The two best matching states are each propagated forward by one step, and the new state is the weighted average of the two image cells. In our investigation of this approach, we found that in most cases similar performance was achieved with simple cell mapping ($k = 1$) and $k = 2$, but that occasionally the $k = 2$ method selected a second best matching unit that was not a good match, and that caused the model error to grow. Thus, for our *relatively coarse* discretization of the state space, we saw no clear advantage to using $k > 1$.

4. Quantification of model reduction error

We demonstrated our model reduction approach in a lattice kinetic Monte Carlo simulation of gallium arsenide deposition, as developed by Itoh and co-workers [5]. Our study included model

reduction, open-loop control via dynamic programming, and a quantification of the errors in the reduced-order models. This study has been submitted for publication in *Automatica* and is included in this report as Appendix A.

The time for the deposition was divided into 10 intervals, and the average modeling error grew as the simulations progressed, as would be expected due to propagation of error from previous steps. We also constructed distribution plots for those errors over hundreds of trajectories, and found that the error distributions are unimodal and approximately Gaussian in shape at each time step. At the end of the 10 steps, the error was similar in magnitude to the error associated with the discretization of the state space. While this is an acceptable and expected level of error, we anticipate that it will continue to grow as the number of steps grows larger and this level of error would become unacceptable. While 10 time steps may seem small from the perspective of simulation and prediction, it does enable the consideration of trajectory planning problems with 10 stages and input levels, which greatly increases the number of possible processing recipes and potential material structures. For example, for our 8 input levels, there are 8^{10} possible trajectories that can be considered. Typically, only constant inputs are used, with only 8 input trajectories possible. Furthermore, exact models over long prediction horizons are not needed when feedback control is used.

Our efforts to quantify and predict the error associated with the model reduction suggested a new approach using spatial statistics [6], which is used in the geology community under the name of kriging. A new project has recently been funded by AFOSR (PI: Gallivan) to develop this method in the context of dynamic models. In the past it has been used extensively but for static models in geology and in mechanical design. It is expected to have wide applicability in error quantification for empirical dynamics models including Markov chains and tabulation models. It

provides a unifying framework to a range of ad-hoc approaches currently used in modeling of a range of Air Force applications including molecular interactions, combustion, and turbulent flow.

5. Control

The majority of the work in this program was on the model reduction approach, since the simple state-affine structure of the final reduced-order model enables the application of most existing methods for dynamic optimization and control. The discrete nature of our state space enabled the implementation of dynamic programming for optimal trajectory generation of the flux profile in the gallium arsenide model. The time-optimal trajectory is reported in Appendix A. We also note that this control approach could be implemented in real-time using dynamic programming, since we can easily compute off-line the time-optimal trajectory for each discrete state in the state space.

The implementation of feedback control requires that the state can be measured or accurately estimated. Estimation and observers were not a primary focus of the work supported by this grant, although we did conduct a study demonstrating that the reduced-order Markov chain model can be used efficiently with an extended Kalman filter, in a simulation study (Gallivan in 2005, *Computers & Chemical Engineering*). Continued work in our group includes a joint modeling and experimental study for estimation of microstructure using limited optical sensors in a chemical vapor deposition process. This work is supported under an NSF grant [7].

6. Modeling of heterostructure field effect transistors

Professor Gallivan spent the summer of 2006 working at Wright-Patterson Air Force Base, under the support of a US Air Force Summer Faculty Fellowship. The goal of the project was to develop a mathematical model for the degradation dynamics of heterostructure field effect

transistors (HFET) made from gallium nitride. Based on the current understanding of hydrogen diffusion in gallium nitride, a mechanism for degradation of AlGa_N/Ga_N high electron mobility transistors was proposed and simulated. Hydrogen is introduced into the devices during their processing and fabrication, and it interacts with defects to electrically passify them. In particular, H⁺ diffuses into the near surface region of the AlGa_N and Ga_N during plasma processing. However, H is the preferred state of the hydrogen interstitial in undoped Ga_N, and thus there is little diffusion of hydrogen during device operation. The traps depassify thermally during operation, which is potentially also enhanced by hot electrons and electromigration. This mechanism would not be reversed by heating, but if H⁺ is present, then heating the device uniformly to 200-300 C would induce recovery of a degraded device. Consideration of the spatial distributions in the HFET is the key to understanding, modeling, and predicting degradation. Modeling of the spatial distributions enables prediction of power law degradation dynamics that are observed in experiments but have not previously been understood theoretically. Experimental validation of this mechanism was needed, and experiments at AFRL are now in progress. Continued interaction with AFRL is planned under the new grant (PI: Gallivan).

7. Modeling of hyperbranched polymers

Structure development in highly branched segmented polyurethaneureas has been investigated by experimental studies and kinetic Monte-Carlo simulations. The experiments were used to develop a minimal set of reaction kinetics for the molecular simulations. Five descriptors were used to describe the state of each polymer (the number of the two types of monomer in the molecule, the number of each type of unreacted end groups, and the number of intramolecular reactions). Additionally the number of molecules of each type is tracked in the simulations. The

simulations have been used to suggest new process trajectories, by changing the feed profile for monomer addition, the dilution of the system, and the amount of monofunctional agents to be added. Two papers have been written, one of which is published (Unal et al. in 2005, *Polymer*) and another which has been submitted and is included as Appendix B of this report. This work has leveraged funding from the Army Research Office through a collaboration with Professor Tim Long at Virginia Tech. Work has focused on validating the model, so the model reduction framework has not yet been applied in this system. However, part of this ongoing collaboration includes the identification of a minimal state representation for molecular architecture in hyperbranched polymers, which is now an unsolved problem. The hyperbranched modeling work has impacted the model reduction methodology in a more indirect way, by providing an alternative, and very different example, for which the model reduction methodology must also be applicable.

III. CONCLUSIONS AND RECOMMENDATIONS

In conclusion, the most significant accomplishment of this research program was the development of an automated model reduction procedure for atomic-scale many body simulations. The reduction was demonstrated in a simulation of gallium arsenide surface processing, and enabled the computation of a time-optimal process trajectory to reach a desired surface structure. The computation required for generation of the model and for the open-loop dynamic optimization was tractable, and enabled a dynamic optimization that would not have been possible using the full stochastic simulations. Additional materials processing applications considered were heterostructure field effect transistors and hyperbranched polymers, and these applications also guided the development of our general model reduction framework. This

framework should have wide applicability in a range of nanostructured materials processing problems.

Journal publications

1. M. A. Gallivan, "An estimation study for control of a lattice model of thin film deposition," *Computers and Chemical Engineering*, special issue on "Simulation and Control of Multiscale Systems," **29** (2005) 761-769.
2. "A data-driven approach for reduction of molecular simulations," C. Oguz and M. A. Gallivan, *International Journal of Robust and Nonlinear Control*, **15** (2005) 727-743.
3. "Understanding the structure development of hyperbranched polymers prepared by oligomeric A₂ + B₃ approach: comparison of experimental results and simulations," S. Unal, C. Oguz, E. Yilgor, M. Gallivan, T. Long, I. Yilgor, *Polymer*, **46** (2005), 4533-4543.
4. "Optimization of a thin film deposition process using a dynamic model extracted from molecular simulations," C. Oguz, and M. A. Gallivan, submitted to *Automatica*. (Included in this report as Appendix A.)
5. "Role of endcapping reactions and the steric hindrance in the synthesis of highly branched poly(ether ester)s via cyclization-free melt condensation of A₂ oligomers and B₃ monomers," C. Oguz, S. Unal, T. E. Long, and M. A. Gallivan, submitted to *Macromolecules*. (Included in this report as Appendix B.)

Peer-reviewed conference proceedings

1. M. A. Gallivan, "Low-order dynamics in a lattice model of thin film deposition, using nonlinear principal component analysis," in *Proceedings of Dynamics and Control of Process Systems*, Boston, July 2004.
2. "Identification of a dynamic model for a thin film deposition process using neural nets," C. Oguz and M. A. Gallivan, *Proceedings of the IEEE International Joint Conference on Neural Networks*, Vancouver, Canada, July 2006.
3. "Identification and Evaluation of a Dynamic Model for a Thin Film Deposition Process," C. Oguz and M. A. Gallivan, accepted for the *Proceedings of the 2007 American Control Conference*.

Personnel supported

- Martha Gallivan (faculty)
- Cihan Oguz (graduate student): May 2004 – March 2007. Developed tools and approach for model reduction and supported polymer modeling
- Jonathan Rawlston (graduate student): January – August 2006, January-March 2007. Supported polymer modeling

Awards

- NSF CAREER award, 2004
- W. David Smith, AIChE CAST, 2005
- US Air Force Summer Faculty Fellowship, 2006

AFRL Point of Contact

- Dr. Donald Dorsey, AFRL/ML, WPAFB, Oh, Phone (937) 255-2227 x3505 (donald.dorsey@wpafb.af.mil). Visited Wright-Patterson Air Force Base in the summer of 2005, gave a seminar. Spent summer of 2006 at WPAFB with Dr. Donald Dorsey through the USAF Summer Faculty Fellowship program. Worked in the Sensor Materials Branch as a part of the PACE program, whose goal is to design electronic devices by incorporating fundamental understanding and mathematical modeling, using a systems engineering approach.

Transitions

- Technology Assist: AFRL. Description: Provided model and code to AFRL/MLPS, of hydrogen dynamics in III-V transistor degradation. Application in PACE (Physics and Chemistry of Electronics). POC: Donald Dorsey.
- Technology Assist: UTRC. Description: The discussions at UTRC helped in shaping the procedure to learn the dynamics of a complex system through short simulations once the appropriate coarse variables have been identified. Application in UTRC's Robust Uncertainty Management program (funded by DARPA). The goal of the RUM Program is to develop new mathematical tools that enable the design of large nonlinear systems with guaranteed performance in the presence of errors. POC: Andrzej Banaszuk.

Appendix

- A. Submitted to *Automatica*: "Optimization of a thin film deposition process using a dynamic model extracted from molecular simulations," C. Oguz, and M. A. Gallivan
- B. Submitted to *Macromolecules*: "Role of endcapping reactions and the steric hindrance in the synthesis of highly branched poly(ether ester)s via cyclization-free melt condensation of A₂ oligomers and B₃ monomers," C. Oguz, S. Unal, T. E. Long, and M. A. Gallivan

References

- [1] M. A. Gallivan and R. M. Murray, "Reduction and identification methods for Markovian control systems, with application to thin film deposition," *International Journal of Robust and Nonlinear Control*, vol. 14, pp. 113-132, 2004.
- [2] G. Berkooz, P. Holmes, and J. L. Lumley, "The proper orthogonal decomposition in the analysis of turbulent flows," *Annual Reviews of Fluid Mechanics*, vol. 25, pp. 539-575, 1993.
- [3] T. Kohonen, "The self-organizing map," *Proceedings of the IEEE*, vol. 78, pp. 1464-1480, 1990.
- [4] C. S. Hsu, *Cell-to-Cell Mapping: A Method for Global Analysis for Nonlinear Systems*. New York: Springer-Verlag, 1987.
- [5] M. Itoh, "Atomic-scale homoepitaxial growth simulations of reconstructed III-V surfaces," *Progress in Surface Science*, vol. 66, pp. 53-153, 2001.
- [6] A. S. Goldberger, "Best linear unbiased prediction in the generalized linear regression model," *Journal of the American Statistical Association*, vol. 57, pp. 369-375, 1962.
- [7] R. Xiong, P. J. Wissmann, and M. A. Gallivan, "An extended Kalman filter for in situ sensing of yttria-stabilized zirconia in chemical vapor deposition," *Computers & Chemical Engineering*, vol. 30, pp. 1657-1669, 2006.

Optimization of a thin film deposition process using a dynamic model extracted from molecular simulations^{*}

Cihan Oguz^a, Martha A. Gallivan^a

^a*School of Chemical & Biomolecular Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA*

Abstract

This study presents and demonstrates an algorithm for computing a dynamic model for a thin film deposition process. The proposed algorithm is used on high dimensional Kinetic Monte Carlo (KMC) simulations and consists of applying principal component analysis (PCA) for reducing the state dimension, self organizing map (SOM) for grouping similar surface configurations and simple cell mapping (SCM) for identifying the transitions between different surface configuration groups. The error associated with this model reduction approach is characterized by running more than 1000 test simulations with highly dynamic and random input profiles. The global error, which is the normalized Euclidean distance between the simulated and predicted states, is found out to be 0.006 on average for the test simulations. This indicates that our reduced order dynamic model, which was developed using a rather small simulation set, was able to accurately predict the evolution of the film microstructure for much larger simulation sets and a wide range of process conditions. Minimization of the deposition time to reach a desired film structure has also been achieved using this model. Hence, our study showed that the proposed algorithm is useful for extracting dynamic models from high dimensional and noisy molecular simulation data.

Key words: Dynamic Modelling; Model reduction; Modelling errors; Optimal control; Order reduction.

1 Introduction

Thin film deposition is a critical step in manufacturing integrated circuits and MEMS devices. As device size gets smaller, film thickness and tolerance approach the atomic scale, so high quality thin films with uniform and smooth surfaces are desired for high device performance. Surface processing is commonly used in developing integrated circuits in order to build features with dimensions of 100 nm and below [1,2]. Some other applications of surface processing are mechanical coatings [3], thermal coatings [4] and MEMS devices [5]. The structure of a thin film is usually a strong function of process inputs such as pressure, plasma power and temperature. Therefore, process models that describe the relationship between inputs and outputs are needed to control the microstructure of thin films. Since continuum assumptions are not valid at such small scales, developing accurate continuum models from material and energy balances

is not possible. An alternative approach is simulating the dynamics of the system by means of molecular simulations, such as molecular dynamics (MD) and Monte Carlo (MC) simulations. In MD simulations, Newton's equations of motion are solved for the position of each atom in a system. Given the initial location of each atom, a potential energy function is used to compute the interaction between atom pairs and this information allows the computation of the trajectory of each atom over a time interval. On the other hand, Monte Carlo (MC) simulation is a stochastic method which generates different discrete configurations of a system by randomly changing the position, orientation and conformation of the atoms or molecules in the system.

The specific application considered in this paper is the epitaxial growth of gallium arsenide (GaAs). GaAs is generally grown by ultra-high vacuum molecular beam epitaxy (MBE). There are many advantages of using GaAs over silicon. It has a higher saturated electron velocity and higher electron mobility, allowing devices to function at frequencies excess of 250 GHz. Also, GaAs devices generate less noise than silicon devices. Having a direct band gap, GaAs can be used to emit light unlike silicon, which has an indirect band gap and is a poor light emitter. MBE deposition of GaAs occurs with a rate of

^{*} This paper was not presented at any IFAC meeting. Corresponding author M. A. Gallivan. Tel. +1 404-894-2878. Fax +1 404-894-2866.

Email addresses: cihan.oguz@chbe.gatech.edu (Cihan Oguz), martha.gallivan@chbe.gatech.edu (Martha A. Gallivan).

approximately one layer per second. Hence, the growth morphology in this process develops in the order of seconds. Even with a further increase in computer power, such a time scale will not be accessible to simulations using MD. Even though MD can capture the atomic vibrations in the order of picoseconds, slower events like an atom overcoming an energy barrier and moving to a new site in the crystal lattice will be infrequent when this technique is used. However, the slower events will be dominant in terms of capturing the dynamics of the process. Therefore, we use Kinetic Monte Carlo (KMC) simulations to reach the time scales of these slower events.

MC simulations are typically used to compute equilibrium properties. In contrast, Kinetic Monte Carlo (KMC) simulations are also able to describe evolution in time. The KMC algorithm was first proposed by Bortz et al. [6] and then transformed by Maksym [7] to be used in MBE growth. It is based on the assumption that the surface evolution is the result of a series of discrete events. These events are classified as adsorption (atom attaching to the surface), surface diffusion (atom hopping) and desorption (atom leaving the surface). In a KMC simulation, the initial arrangement of atoms is specified and the possible transitions from this configuration are evaluated. Then, based on generated random numbers, one transition and its location is selected. After the transition is executed, the simulation clock is incremented using another random number. A KMC simulation model on a zincblende crystal structure has been developed to describe epitaxial growth of GaAs [8]. This model includes the rates (derived from experiments) of over one thousand possible events taking place on the surface.

Several approaches have been developed in the past few years to integrate molecular simulations with dynamic analysis and optimization. One approach is the derivation of stochastic time evolution equations from the probabilistic master equation [11], under assumptions that are mostly applicable for well-mixed reacting systems. An alternative approach for constructing stochastic differential equations from molecular simulations of film growth is the generation of stochastic spatially distributed PDEs for the height profile of a surface [12]. These models describe continuum behavior and dynamics, and thus are not appropriate for modeling surface structure at atomic scales. This modeling approach, with stochastic PDEs, has been used recently to control the surface roughness in a thin film deposition process on a one-dimensional lattice [13]. On the other hand, controller design based on molecular simulations is an active area of research [14]. In a recent study [15], a reduced order stochastic model obtained from KMC and finite difference simulation data has been used to design a feedback-feedforward controller in order to maintain the current density during the copper electrodeposition process at a constant level.

In order to explicitly model and predict discrete atomic scale structure, a model is required that does not average over the small length scales. One approach to address this issue is equation-free computing, which was first used for stability and bifurcation analysis [9]. A low-dimensional system state was assumed (based on macroscopic arguments), and short simulations were run from specific initial conditions to approximate time-derivatives. More recently, this method has also been applied for optimization [10]. However, the reduction in computational time achieved by this method, compared to running full molecular simulations, may not be sufficient to make the approach practical when many predictions over long times intervals are required.

In an alternative approach presented by Gallivan and Murray [16], molecular simulations were used to construct Markov models, with discrete states describing groups of similar configurations. This grouping strategy was based on the similarity of the roughness values, and enabled computation of the optimal temperature profile by penalizing the surface roughness and temperature changes during the thin film deposition. The construction of this explicit low-order model reduced the computational load by four orders of magnitude, when compared with full molecular simulations. This significant reduction was critical for making the dynamic simulation and optimization feasible over macroscopic processing times.

The construction of a dynamic model relies on the existence and knowledge of the system state. While a low-order state was selected in previous studies using physical arguments [9,16], the detailed structure in a molecular simulation may not even have a low-order representation. Additionally, small structural features may have a large effect on the time evolution of the system. In order to address this issue, a previous study by Oguz and Gallivan [17] proposed the use of high-dimensional step-step correlation functions, which provided a more detailed state description of the film surface, compared to the surface roughness alone. This study demonstrated the implementation of principal component analysis (PCA) for reducing the state dimension and self organizing map (SOM) for automated grouping of the similar states in the state space. In a more recent study, Varshney and Armaou [18] also used spatial correlation functions to characterize the state of their film growth simulation, and used equation-free computing to simulate the dynamics [9].

The next section explains our modeling approach, which consists of state reduction and model identification. State reduction is achieved by reducing the dimension of the simulation data using principal component analysis (PCA) and grouping similar surface configurations using a self organizing map (SOM). After establishing a discrete state space, transitions between different surface configuration groups have been identified by simple

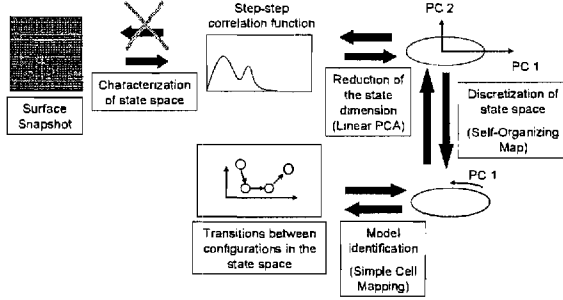


Fig. 1. Schematic of the modeling approach

cell mapping (SCM) for model identification purposes. In Section 3, we evaluate the performance of the process model by computing the local and global prediction error under highly dynamic input profiles. Section 4 describes the minimization of the film deposition time to reach an optimal film structure. Finally, Section 5 provides the conclusions of this study.

2 Modeling Approach

Our modeling approach, which consists of four steps, is illustrated in Figure 1. Characterization of the state space, reduction of state dimension, discretization of the state space, and model identification gives us a reduced order dynamic model.

2.1 Characterization of the state space

A step-step correlation (SSC) function is used to describe the microscopic state of the system during the simulations. This function gives the distance and orientation between pairs of steps on the surface, where a step is defined as a change in height from one atomic surface site to the next. Since the occurrence of every type of step pair (up-up, up-down, down-up and down-down) is counted on the surface in both directions on the Cartesian coordinate axes, the spatial correlation function is high-dimensional and may contain redundant information. The data is also noisy. Noise is reduced by performing multiple realizations under identical conditions and averaging the results. After averaging, PCA is used to determine the number of independent variables needed to fully determine the SSC function. This technique is widely used to eliminate linear correlations among the variables in data sets. It is a crucial step in our study. The reduced state dimension makes it possible to construct a compact dynamic model.

The KMC simulations have been carried out using the kinetic barriers calculated by Itoh [8] with the following parameters:

- Growth temperature: 580 C
- Film deposition interval: 0.20 monolayers (ML)

- Incident arsenic dimer (As_2) flux: 0.4 ML/s
- Incident gallium (Ga) flux: Varied between 0.06-0.20 ML/s (0.06, 0.08, 0.10, ..., 0.20) flux range where the model is valid.
- Lattice size: 300x300 (90000 surface atoms) with periodic boundary conditions

The starting surface in our simulations is the thermodynamic ground state of GaAs(001), the $\beta 2(2 \times 4)$ reconstruction which prevails in a wide range of growth conditions.

Snapshots of surfaces have been recorded at surface increments of 0.01 ML, starting from an initial surface in the $\beta 2(2 \times 4)$ configuration, up to 0.20 ML coverage. Eight constant input simulations were performed at Ga fluxes of 0.06 ML/s, 0.08 ML/s, 0.10 ML/s, ..., 0.20 ML/s and we will refer this as Training Simulation Set 1. In Training Simulation Set 2, we again have 8 simulations, but this time the flux is kept constant up to 0.10 ML coverage (middle of the deposition), and the flux is shifted to a different value at that coverage point. To explore the KMC state space even further, we perform 60 additional simulations (Third Training Simulation Set), where two flux shifts are made at 0.07 and 0.14 ML coverage points.

The SSC function has been computed as $\mathbf{S} \in \mathbb{R}^{d_s}$ for each snapshot. \mathbf{S} is obtained by combining the 16 different portions of the SSC function, which has been evaluated for four different types of step pairs in four different directions on the surface ($4 \times 4 = 16$ combinations). Each portion has 300 variables, which is equal to the lattice size. Therefore, d_s has a value of 4800 ($300 \times 16 = 4800$). We have a total number of snapshots $n_s = 1521$ (76 simulations, 20 snapshots for each simulation, plus the snapshot of the initial state of the system). \mathbf{S} of all snapshots in the training data are collected in $D \in \mathbb{R}^{4800 \times 1521}$. Before performing PCA, D is first transformed into $D' \in \mathbb{R}^{4800 \times 1521}$ with the following elements:

$$D'_{i,j} = D_{i,j} - D_{mean,i}. \quad (1)$$

Here, $D_{mean} \in \mathbb{R}^{4800}$ is defined as:

$$D_{mean,i} = \frac{\left(\sum_{j=1}^{n_s} D_{i,j} \right)}{n_s} \quad (i = 1, 2, \dots, 4800). \quad (2)$$

In order to complete the pre-processing of D , D' is transformed into $D'' \in \mathbb{R}^{4800 \times 1521}$.

$$D''_{i,j} = D'_{i,j} / D_{std,i} \quad (i = 1, 2, \dots, 4800, j = 1, 2, \dots, 1521) \quad (3)$$

$D_{std} \in \mathbb{R}^{4800}$ is defined as:

$$D_{std,i} = \sqrt{\frac{\sum_{j=1}^{n_s} (D_{i,j} - D_{mean,i})^2}{n_s}} \quad (4)$$

Hence, by pre-processing D , the variance of each variable in the SSC function is made equally important before PCA is carried out. In other words, small features with low variance will not be neglected.

2.2 Reduction of the state dimension

Principal component analysis (PCA) has been performed through the singular value decomposition of matrix D'' by computing the singular values of this matrix. The squares of the singular values correspond to the eigenvalues of the covariance matrix of D'' , and the ratio of each eigenvalue to the sum of all eigenvalues (normalized eigenvalue) is plotted against the principal components. The point on the plot, where a sudden decay of the normalized eigenvalue is seen, gives the minimum number of principal components (the minimum dimension n) that can reconstruct the data effectively. At this point, we project each snapshot's SSC function onto the first n principal components D'' . We define $\mathbf{x} \in \mathbb{R}^n$ as the coefficient set with these new coordinates. Each \mathbf{x} characterizes a particular snapshot. As a result of PCA, small features in the SSC function, which do not contribute to the variance of the D'' significantly, are eliminated. We note that this could create a problem later while grouping similar surface structures according to first n principal components, because these small features may possess valuable information about the differences between the surface structures and their evolution in time.

2.3 Discretization of state space

After performing PCA and obtaining a coefficient set to characterize each surface snapshot, we use self organizing map (SOM) to eliminate some of the nonlinear correlations within the data set by employing MATLAB's SOM Toolbox. Snapshots with very similar microstructures are grouped by SOM, and snapshots in the same group are viewed as equivalent when identifying the dynamic model in the next step. The PCA data is discretized as a part of this step, since snapshots are grouped among the nodes of SOM. As a result, the number of discrete states is finite. The computational load for the model identification is also reduced with the grouping achieved by

SOM. Because, during model identification, rather than the transitions between each surface structure, only the transitions between the groups are computed.

Various training procedures for SOM are described elsewhere [19]. SOM training is accomplished by sequential or batch-wise comparison of snapshots to the overall map, and then updating the prototype vectors $\mathbf{v} \in \mathbb{R}^n$ of the map nodes to match and organize the snapshots. Once the map is trained, each snapshot is associated with the node that has the closest prototype vector, as measured by the Euclidean distance. The default number of nodes in the map is computed by the SOM toolbox using a heuristic formula which is a function of the number of rows (snapshots) in the data matrix. The map size can also be increased to provide a finer discretization of the state space. Another important SOM parameter, the ratio of the sidelengths of the map, is set equal to the ratio of the two largest principal components of the data matrix.

2.3.1 Results

In this study, PCA is used to find the minimum dimension that can represent the microscopic state of the surface snapshots recorded during the simulations. This method consists of computing the variance captured by each principal component of the entire data set and selecting the most important principal components that can reconstruct the data well. Figure 2 shows the normalized eigenvalues of each principal component. A knee shape is observed after the second principal component and the first two principal components capture nearly all of the variance within the data set.

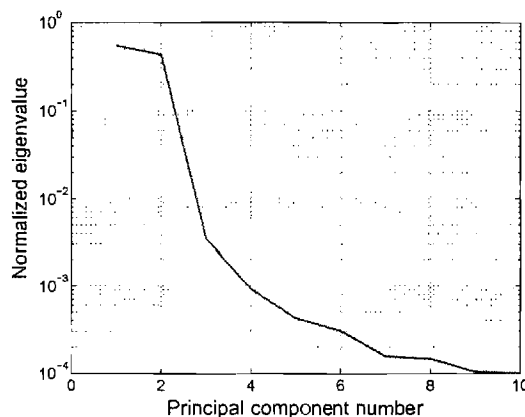


Fig. 2. Normalized eigenvalues versus principal components

However, data reconstructions with 2 and even more principal components showed us that at least 5 components are needed in order to effectively represent our data. Specifically, 5 principal components reconstructed the small clusters of atoms on the surface (with a size of

less than 20 lattice units) much better than 2 principal components. Figure 3 illustrates this comparison.

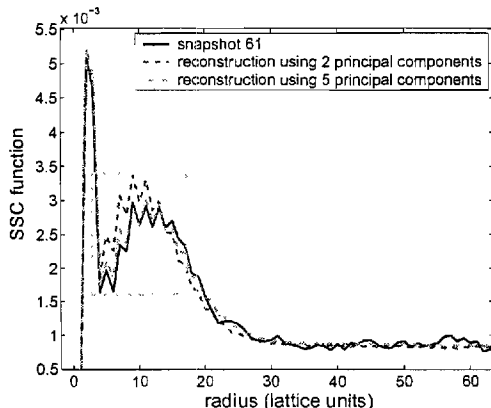


Fig. 3. Comparison of the reconstructions using 2 and 5 modes with the original data for snapshot 61. This surface is obtained under 0.10 ML/s Ga flux and has a 0.20 ML surface coverage. The region enclosed with the gray dashed line represents the surface structures with a size of less than 20 lattice units.

As already defined in Section 2.2, the coefficient set \mathbf{x} for each surface snapshot is obtained by taking the inner product of the snapshot’s SSC function with the $n = 5$ principal components of the data matrix D'' . This reduced representation, which includes the coefficient sets of every snapshot, is used to train an SOM. For training, we used the default map size, which is based on the number of surface snapshots in the training data. Each SOM node has its own prototype vector having the same dimension as each snapshot in the data set ($n = 5$). As a result, each map node also has its own SSC function S , which is the PCA reconstruction of its prototype vector \mathbf{v} . Thus, each map node is associated with a type of surface.

The quality of the resulting map can be evaluated by calculating the average quantization error over the input data, given by $\|\mathbf{x} - \mathbf{v}_{BMU}\|_2$, where \mathbf{v}_{BMU} is the prototype vector for the best matching unit, for the snapshot with the projection vector \mathbf{x} [19]. In words, the Euclidean distance between each surface snapshot and its best matching SOM node is computed, and the average of these distances gives the average quantization error.

In order to see the effects of training data on the map quality, we generated multiple maps while keeping the map size at 192 nodes. For each map, the training phase has taken approximately 3 minutes using a Pentium 4 processor with a speed of 3 GHz. The first map SOM1 is trained with 161 surface snapshots (from Training Simulation Set 1), the second map SOM2 is trained with 321 surface snapshots (from Training Simulation Sets 1 and 2) and SOM3 is trained using all of the 1521 surface snapshots (from Training Simulation Sets 1, 2 and

3). Table 1 shows the statistics of three maps with 192 nodes, which were trained using these three different data sets. SOM1 and SOM2 have similar statistics, except that SOM2, which has been trained with a larger data set, has a larger number of surface snapshot groups. This suggests that Training Simulation Set 2, where the flux was changed in the middle of the simulations at 0.10 ML surface coverage, enabled us to see surface structures that had not been accessible through constant input simulations (Training Simulation Set 1). On Table 1, as we move from SOM2 to SOM3, we again observe an increase in the number of surface snapshot groups. Therefore, we conclude that Training Simulation Set 3 (during which the flux has been changed twice during the simulations) explored the KMC state space even further, producing additional surface structures compared with Training Simulation Sets 1 and 2. SOM3 has an average quantization error of 5.1, whereas the average prototype vector size is 53.0. If needed, the average quantization error can be reduced by adding more nodes to the map, but we were able to obtain a useful model with SOM3.

Another SOM metric is the topographic error, which is the proportion of all the data vectors for which first and second best matching units are not adjacent on the map [19]. SOM3 also has a topographic error of 0.01, which means that for only 15 of the 1521 snapshots, the first and second best matching units were not adjacent on the SOM. Topology preservation is not required for a good state representation, but it aids in visualization of the dynamics on the map.

At this point, three SOMs have been generated. Only one of them, SOM3, will be used to identify a dynamic model, since it has been trained with the most extensive amount of training data.

Table 1
Statistics of three SOMs trained using different data sets

Map	Training sim. sets	Quant. error	Proto. vec. size	Topog. error	Number of groups
SOM1	1	4.1	54.1	0.01	119
SOM2	1, 2	4.8	54.4	0.02	142
SOM3	1, 2, 3	5.1	53.0	0.01	175

2.4 Model Identification

2.4.1 Simple Cell Mapping

Characterization of the surface snapshots from all of the training simulations gives the system’s state space. As explained in the previous section, the surface snapshots are grouped into discrete states using SOM, according to their similarities in terms of surface morphology. These groups represent the cells, and simple cell mapping (SCM) or generalized cell mapping (GCM) can

be used for obtaining a global view of the behavior of the system [21]. A cell map is formed by dividing the state space into a finite number of cells (using SOM) and approximating the behavior of the system by means of transitions between the cells. SCM is a deterministic approach in which each cell is mapped into exactly one other cell for a particular input. In GCM, each cell can have several image cells. In other words, a cell can be mapped to several other cells. Using this stochastic approach, we can assign a probability of the system being in a cell at a specific time and extract the dynamic properties. In the current study, we implement the SCM approach to identify the dynamic model for the process, because SCM provides a deterministic model to describe the evolution of the surface structure under different material flux profiles. The following are the steps used for model identification:

- One surface configuration is selected from a particular map node.
- A new simulation is started from that surface configuration and run for an incremental coverage interval with one of the flux settings.
- The final surface structure is recorded and the configuration group, which has the closest microstructure to that final structure, is obtained. In other words, the flux-dependent transition (from one configuration group to another) is found.

When the procedure described above is repeated for every map node and each flux setting, we identify a reduced order dynamic model. The identification procedure requires running $146 \times 8 = 1168$ KMC simulations (for 146 map nodes and 8 flux settings), which takes approximately 7.3 days using a computer cluster made up of 16 computers, each having an Intel Xeon processor with a speed of 2.66 GHz.

The reduced order model is described as:

$$\mathbf{p}[j+1] = A(u[j])\mathbf{p}[j] \quad (5)$$

where j is the time step (or coverage level) number, $u[j] \in \mathbb{R}$ is the input (or flux) value, $\mathbf{p} \in \mathbb{R}^m$ is the probability vector describing the configuration group that the system is currently in and $A \in \mathbb{R}^{m \times m}$ is the transition matrix. The value of m (dimension of \mathbf{p}) is equal to the number of configuration groups in the state space (the number of cells in the cell map). We define \mathbf{p}_i as:

$$\mathbf{p}_i = \begin{cases} 1 & \text{if } i = l; \\ 0 & \text{if } i \neq l. \end{cases}$$

Here, l is the number of the current system configuration. The transition matrix A is a function of the input.

Therefore, in the reduced order model, we have eight transition matrices for eight flux settings. The sum of the elements in each column of A is equal to one, because for each column, only one element of A is non-zero, whereas the other elements are equal to zero. In other words, there is only one possible transition from each configuration group under a particular input. This construction of A is due to the deterministic nature of the simple cell map. If the system makes a transition from configuration s to t at time step j with the flux setting $u[j]$, this transition can be represented by a transition matrix with the following properties:

$$A_{k,s} = \begin{cases} 1 & \text{if } k = t; \\ 0 & \text{if } k \neq t. \end{cases}$$

At time step j , the surface properties of the system are given by $\mathbf{x}[j]$ (set of projection coefficients defined in Section 2.2):

$$\mathbf{x}[j] = X\mathbf{p}[j] \quad (6)$$

where $X \in \mathbb{R}^{n \times m}$ has the $\mathbf{v} \in \mathbb{R}^n$ (prototype vector) of each configuration group (SOM node). In other words, \mathbf{v} of the configuration group i is $\mathbf{v}_i = \{X_{1,i}, X_{2,i}, X_{3,i}, X_{4,i}, X_{5,i}\}$.

2.4.2 K-nearest neighbor algorithm

Cell mapping can be generalized to enable interpolation between cells. In order to carry out the interpolation, we use the k-nearest neighbor algorithm [20] and predict the evolution of \mathbf{x} for a given flux profile. When $k=1$, this algorithm produces results identical to simple cell mapping and no interpolation is made. The following steps represent the k-nearest neighbor algorithm when $k=2$:

- (1) Let \mathbf{x}_{old} be the vector representing the surface state at an initial film coverage. This vector is compared with all the prototype vectors on the SOM and its best matching unit (BMU) and second BMU are found based on the Euclidean distances between these vectors. Let these two SOM nodes have prototype vectors \mathbf{v}_1 and \mathbf{v}_2 , where \mathbf{v}_1 and \mathbf{v}_2 represent the first BMU and second BMU, respectively.
- (2) Compute the distances $d_1 = \|\mathbf{x}_{old} - \mathbf{v}_1\|_2$ and $d_2 = \|\mathbf{x}_{old} - \mathbf{v}_2\|_2$.
- (3) Compute the weights associated with these two distance values $w_1 = (1/d_1)/(1/d_1 + 1/d_2)$ and $w_2 = (1/d_2)/(1/d_1 + 1/d_2)$.
- (4) Predict the system state at the next step from $\mathbf{x}_{new} = w_1 \cdot \mathbf{v}_3 + w_2 \cdot \mathbf{v}_4$, where \mathbf{x}_{new} is the state vector prediction for the next step, \mathbf{v}_3 and \mathbf{v}_4 are the prototype vectors of the nodes determined from

SCM (v_1 transitions into v_3 , and v_2 transitions into v_4 under the particular flux setting).

- (5) Set $\mathbf{x}_{old} = \mathbf{x}_{new}$ and repeat steps 2-4 to make predictions at higher film coverage values. At each coverage value, the predicted \mathbf{x} can be converted to the high dimensional SSC function by multiplying the elements of \mathbf{x} with the eigenvectors obtained from the training data D'' .

The algorithm described above is the k-nearest neighbor algorithm with $k=2$. Here, the interpolation between prototype vectors provide the opportunity to improve the accuracy of the prediction. On the other hand, when $k=1$, k-nearest neighbor algorithm is identical to SCM. In that case, the second BMU is not identified and the weights are not computed.

3 Performance Evaluation of the Dynamic Model

3.1 Local Error Quantification

The core hypothesis of our model reduction approach is that the surface structures in the same configuration group (SOM node) should show similar dynamic behavior under identical process input (material flux). SCM, which is used to extract a dynamic process model in this study, is also useful for testing this hypothesis. We define the cell mapping error (CME) to quantify the different dynamic behaviors of surface structures which belong to the same configuration group. The following is the procedure we used to find CME :

- (1) Randomly, select one surface structure from a map node and identify where this structure is mapped on the SOM under a particular flux setting (first cell mapping).
- (2) Randomly, select another surface structure from the same map node and run another simulation starting with this new surface structure and perform SCM again (second cell mapping).
- (3) Compute $CME = \|\mathbf{S}_1 - \mathbf{S}_2\|_2 / (\|\mathbf{S}_1 + \mathbf{S}_2\|_2 / 2)$, where \mathbf{S}_1 and \mathbf{S}_2 are the SSC functions of the SOM nodes coming from the first and second cell mapping, respectively. It should be noted that these functions are reconstructed from the prototype vectors of the map nodes.
- (4) Repeat steps 1-3 for all SOM nodes under all flux settings.

CME is computed by picking two different surface structures from each node and performing SCM under each of the eight flux settings. In order to characterize the distribution of CME , its cumulative distribution function is computed. This function is defined as $CDF(y) = P(CME \leq y)$ and gives the value of the probability that $CME \leq y$ for a given y . Figure 4 shows the CDF of the CME . 52% of the mappings are identical, with

both surfaces evolving to the same map node. In those cases, $CME = 0$. Also, with a 0.90 probability $CME \leq 0.0075$, supporting the fact that there is a very strong chance for surface structures in the same group to show similar dynamic behavior. In order to reduce CME , a larger SOM can be used. The tradeoff is that this would increase the dimension of the cell map and the computational load associated with the system identification step.

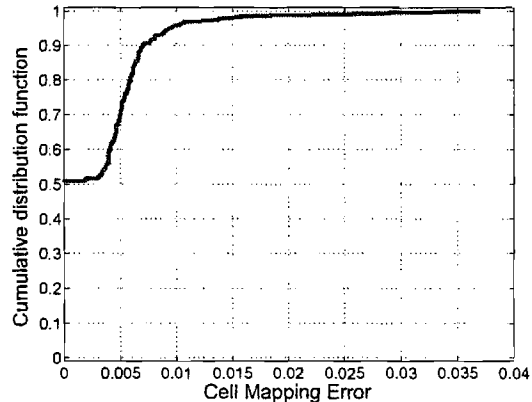


Fig. 4. Cumulative distribution function of the cell mapping error

3.2 Global Error Quantification

A dynamic model is constructed by finding the flux dependent transitions between the nodes of SOM3 under all flux settings. In order to test the predictive ability of our model, 1210 test simulations are performed. In these simulations, we split the film coverage domain (from 0 to 0.20 ML) into 10 equal intervals. The gallium flux value is kept constant in each interval. However, its value randomly changes when moving from one coverage interval to the next. This strategy provides very different flux profiles than the ones used to generate the training data, which involved a maximum of two flux switches. Figure 5 shows the flux profile for one of these simulations during which the flux is randomly changed at each step.

As described in Section 2, surface snapshots from the test simulations at different coverage levels (10 surface snapshots from 0.02 ML to 0.20 ML film coverage) are characterized using SSC functions, and the coefficient set $\mathbf{x} \in \mathbb{R}^n$ of each surface snapshot is computed. Then, each \mathbf{x} is matched with an SOM node based on the criteria of minimum Euclidean distance between \mathbf{x} and the prototype vectors of the SOM nodes. In other words, the best matching unit for each \mathbf{x} is sought. For the test simulation with the flux profile shown in Figure 5, the predicted trajectory and the KMC simulation trajectory are given in Figure 6. Here, the prediction comes from the k-nearest neighbor algorithm (with $k=1$) and each hexagon represents an SOM node corresponding to a

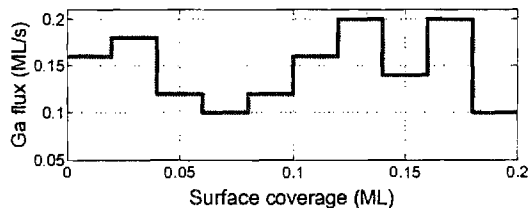


Fig. 5. Flux profile of a test simulation

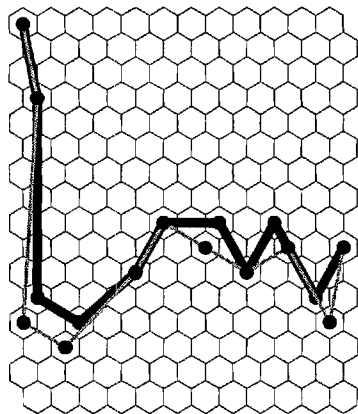


Fig. 6. Trajectories of the KMC test simulation (red line) and the prediction (black line) on SOM3

surface structure group. The simulation follows a path starting from the SOM Node 1 (initial surface structure) and moves to the right hand side of the SOM as the film coverage increases. Since each simulation starts from SOM Node 1 and the transitions from each node under all flux settings are known, the prediction of the simulation trajectory (using the dynamic model) is straightforward. Figure 6 indicates that there is an agreement between the predicted trajectory and the trajectory of the KMC simulation. Similar results were obtained for all test simulations. In some instances, these trajectories passed through neighboring SOM nodes. However, as Table 1 indicates, SOM3 has a very low topographic error. Hence, neighboring SOM nodes are similar, so slight differences between these trajectories do not jeopardize the accuracy of the state predictions.

SOM Node 186 represents the predicted film structure at the final coverage value (0.20 ML). Figure 7 shows the SSC function reconstructed from the prototype vector of this SOM node. The reconstruction agrees with the KMC simulation's SSC function. Hence, our dynamic model is capable of predicting the final film structure quite well. This figure also illustrates that the noise in the KMC simulation data is considerably reduced when the SSC function is reconstructed from the prototype vector.

In order to evaluate the performance of the reduced order dynamic model, we quantify the prediction error for the

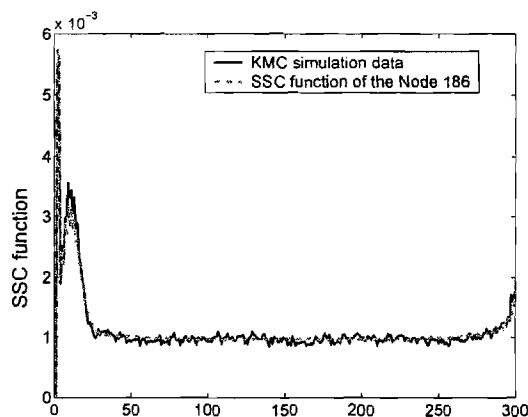


Fig. 7. Reconstruction of the SSC function with the prototype vector of the SOM Node 186 and the original KMC simulation data at final film coverage

test simulations, which were not included in the training data. The global (multi-step) error associated with the predicted SSC function is defined as:

$$E_{SSC} = \|\mathbf{S}_s - \mathbf{S}_p\|_2 / \|\mathbf{S}_s\|_2, \quad (7)$$

where \mathbf{S}_s is computed from the KMC simulation data and \mathbf{S}_p is reconstructed from the predicted state vector \mathbf{v} (prototype vector of the SOM node). The values of E_{SSC} at different film coverage levels (for the test simulation with the flux profile in Figure 5) are less than 0.018 according to Figure 8. This figure compares the error in the predictions made using the k-nearest neighbor algorithm with different k values. For this particular test simulation, predictions are less accurate with k=2. However, for more than 90% of the test simulations, predictions did not change when k value was increased from 1 to 2. This is because, at low film coverage levels, different surface configuration groups (corresponding to different SOM nodes) do not possess significantly different surface features (or different \mathbf{x}) and they are mapped to the same SOM node at subsequent coverage levels. Therefore, throughout the rest of this section, we only report the results obtained with k=1, which is identical to simple cell mapping.

As a part of the global error quantification, we also computed E_{SSC} at nine evenly distributed film coverage values (0.04 ML, 0.06 ML, ..., 0.20 ML) for three large test simulation sets (with random input profiles), none of which had been in the SOM training data. It should be noted that the SOM node 1, which represents the initial film structure at 0 ML, maps to the same SOM node at 0.02 ML with all flux settings. Hence, the prediction error at 0.02 ML is not computed. The three test simulation sets include 300, 600 and 1210 simulations, respectively. Figure 9 compares the mean values of E_{SSC}

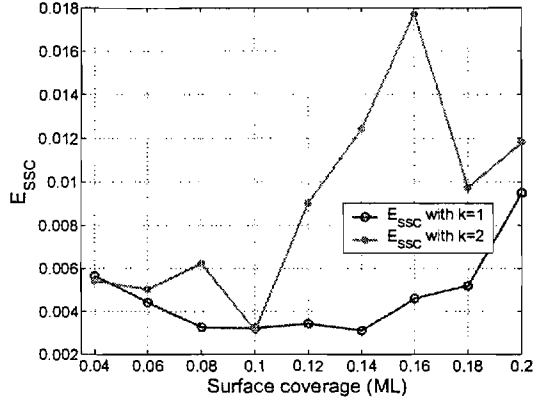


Fig. 8. The evolution of E_{SSC} for the test simulation with $k=1$ and $k=2$

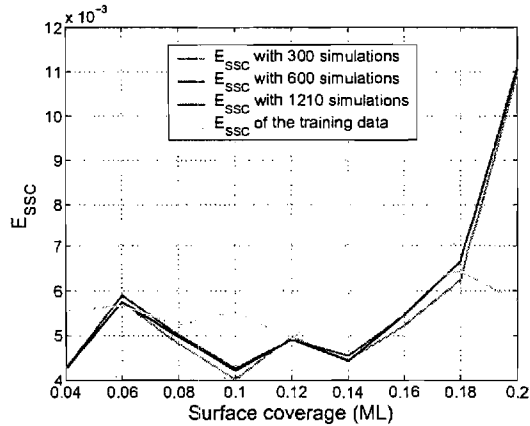


Fig. 9. The mean value of E_{SSC} at different film coverage levels for three test simulation sets and the training data

at different film coverage levels for these three test simulation sets and also the Training Simulation Set 3. The mean E_{SSC} is below 0.012 at all coverage levels. It is interesting to note that the dynamic model was able to predict the final film structure in the training simulations with less error than the test simulations. This is because of the fact that the data from the training simulations were used as an input in the system identification, where as our dynamic model was not as familiar with some of the surface structures produced during the test simulations, especially at high film coverage.

As shown in Table 2, the mean values of E_{SSC} for the predictions associated with the test simulation sets are around 0.006, but the standard deviation values are comparable to the mean values indicating a wide distribution of the prediction error. In order to get a more clear idea about the distribution of E_{SSC} , its cumulative distribution function (CDF) is computed for the three test simulation sets and also the Training Simulation Set 3. Figure 10 shows that the CDF curves of the test sim-

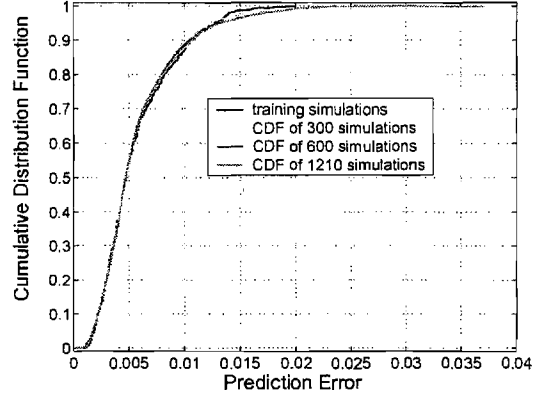


Fig. 10. Cumulative distribution function of the E_{SSC} for three test simulation sets

ulation sets are very similar and enlarging the size of the simulation set does not change the distribution of the prediction error significantly. Also, the probability of having an E_{SSC} less than 0.01 is around 90%, which again supports the fact that the reduced order dynamic model has a good prediction capability.

Table 2

Mean and standard deviation values of E_{SSC} for three test simulation sets

Test simulation set	Mean	Standard deviation
1	0.0057	0.0035
2	0.0058	0.0037
3	0.0058	0.0037

In the last part of this section, we compare different types of prediction error to understand the major factors contributing to the inaccuracies in the model predictions. We define two alternatives to E_{SSC} in order to investigate the effect of normalization (using D_{mean} and D_{std} defined in Section 2.1.) on the prediction error. The first alternative to E_{SSC} is

$$E_{SSC'} = \|\mathbf{S}'_s - \mathbf{S}'_p\|_2 / \|\mathbf{S}'_s\|_2 \quad (8)$$

where \mathbf{S}'_s and \mathbf{S}'_p are the normalized versions of \mathbf{S}_s and \mathbf{S}_p :

$$\mathbf{S}'_{s,i} = \mathbf{S}_{s,i} - D_{mean,i} \quad (i = 1, 2, \dots, 4800) \quad (9)$$

$$\mathbf{S}'_{p,i} = \mathbf{S}_{p,i} - D_{mean,i} \quad (i = 1, 2, \dots, 4800) \quad (10)$$

$D_{mean,i}$ is already defined in Section 2.1. The second alternative to E_{SSC} is:

$$E_{SSC''} = \|S''_s - S''_p\|_2 / \|S''_s\|_2 \quad (11)$$

where S''_s and S''_p are the normalized versions of S'_s and S'_p , respectively:

$$S''_{s,i} = S'_{s,i} / D_{std,i} \quad (i = 1, 2, \dots, 4800) \quad (12)$$

$$S''_{p,i} = S'_{p,i} / D_{std,i} \quad (i = 1, 2, \dots, 4800) \quad (13)$$

$D_{std,i}$ was also defined in Section 2.1. Table 3 shows that mean value of $E_{SSC''}$ (0.2675) is much higher than that of $E_{SSC'}$ (0.1758). Comparison of the CDF of $E_{SSC'}$ and $E_{SSC''}$ given in Figure 11 also shows a higher $E_{SSC''}$ for any value of the CDF. This indicates that features with large variance are predicted more accurately than the ones with small variance. It follows then, that the features with small variance are more noise-corrupted in the KMC simulations.

The quantization error, which is defined in Section 2.3.1, provides a minimum bound on the prediction error. The average quantization error (E_q) for SOM3 is 0.1094. This error is due to the discretization of the state space, and does not include any additional error propagated from one step to the next one through the dynamics of the thin film deposition process. Therefore, we define another kind of prediction error as

$$E_x = \|\mathbf{x}_s - \mathbf{x}_p\|_2 / \|\mathbf{x}_s\|_2. \quad (14)$$

Here, $\mathbf{x}_p \in \mathbb{R}^5$ represents the predicted state and $\mathbf{x}_s \in \mathbb{R}^5$ is computed from the KMC simulation data:

$$\mathbf{x}_{p,i} = S''_p \cdot \mathbf{U}_i^T \quad (i = 1, 2, \dots, 5) \quad (15)$$

$$\mathbf{x}_{s,i} = S''_s \cdot \mathbf{U}_i^T \quad (i = 1, 2, \dots, 5) \quad (16)$$

where $\mathbf{U}_i \in \mathbb{R}^{4800}$ is the i^{th} principal component of D'' .

As shown in Table 3, the mean value of E_x , which is computed from the prediction of states in Test Simulation Set 3, is twice as high as E_q incurred during the training of SOM3. From this result, it can be concluded that the propagation error due to dynamics is not negligible and accounts for approximately half of E_x . Hence, the other half of the prediction error comes from the discretization of the state space. Figure 11, which compares the CDF of E_x and E_q , also supports this conclusion. Here, for any particular value of error, the probability of having that particular value of E_x is higher than that of E_q .

Table 3

Mean values of E_q , E_x , $E_{SSC'}$ and $E_{SSC''}$. E_q , which is computed for each data vector and the prototype vector of the data vector's best matching unit on SOM, is the difference between data vector and prototype vector during SOM training. The other errors are the normalized versions of E_{SSC} .

Error type	Mean
E_q	0.1094
E_x	0.2277
E_{SSC}	0.0057
$E_{SSC'}$	0.1758
$E_{SSC''}$	0.2675

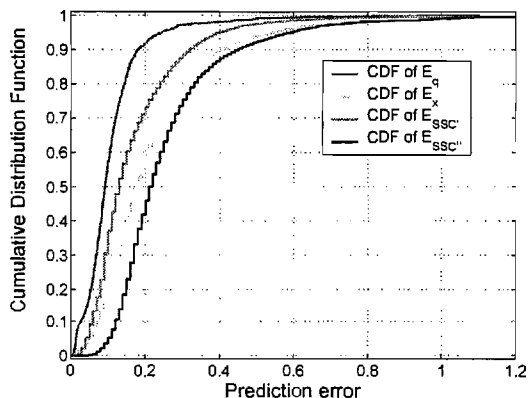


Fig. 11. Cumulative distribution function of different types of error

4 Optimization of the final film structure and the deposition time

In this section, our dynamic model is used to minimize the deposition time to reach a particular surface configuration. One desired structure is a very regular surface with many large islands and a very low Ga adatom (isolated Ga atom) density. This surface structure can be identified by minimizing:

$$F = a_i - b_i c_i \quad (17)$$

where a_i , b_i , and c_i are the values of Ga adatom density, typical island size and the number of islands with the typical island size for surface configuration i , respectively. From the Training Simulation Sets 1, 2 and 3, we extract a_i , b_i , and c_i values of the surfaces at 0.20 ML surface coverage.

According to equation (17), the optimal surface structure is Snapshot 861 (from Simulation 43). Snapshot 861 is matched with the Node 182 of SOM3 during the training. This node is accessible through a constant input simulation with a flux of 0.08 ML/s. Identification of the

flux profile that would reach Node 182 in the minimum amount of time can be posed as a dynamic programming problem [22].

Let the decision variables $d_q (q = 1, 2, 3, \dots, 11)$ be the immediate destinations on different stages. In this problem, we have 11 stages for 11 film coverage levels (0 ML, 0.02 ML, 0.04 ML, ..., 0.20 ML) and d_q corresponds to the map node number at stage q . Thus the route (trajectory) of the deposition is $d_1, d_2, d_3, \dots, d_{11}$, where $d_1 = 1$ and $d_{11} = 182$ since the initial surface structure is represented by SOM Node 1 and the final (optimal) surface structure is in Node 182.

Let $f_q(s, d_q)$ be the total cost of the best overall policy for the remaining stages, given that we are in state s (number of the map node we are currently in), ready to start stage q , and d_q (the number of the map node we are moving to) is selected as our immediate destination. Here, the total cost is the deposition time and each deposition interval is 0.02 ML long. Given s and q , let d_q^* denote any value of d_q that minimizes $f_q(s, d_q)$ and let $f_q^*(s)$ be the corresponding minimum value of $f_q(s, d_q)$. Thus,

$$f_q^*(s) = \min_{d_q} f_q(s, d_q) = f_q(s, d_q^*) \quad (18)$$

where

$$f_q(s, d_q) = c_{s, d_q} + f_{q+1}(s, d_{q+1}) \quad (19)$$

Here, the cost c_{s, d_q} , is the time incurred while moving from s to d_q , given by:

$$c_{s, d_q} = L/F_{s, d_q}, \quad (20)$$

where L is the length of a single coverage interval (0.02 ML) and F_{s, d_q} is the value of the gallium flux that provides a transition from state s to d_q . The ultimate destination reached at the end of stage 11, $f_{11}^*(182) = 0$. The objective is to find $f_1^*(1)$ and the corresponding route. Dynamic programming can solve this problem by successively finding $f_{10}^*(s)$, $f_9^*(s)$, $f_8^*(s)$, ..., $f_2^*(s)$ and using $f_2^*(s)$ to solve for $f_1^*(s)$. This is achieved by eliminating some of the suboptimal trajectories as we move from $f_{10}^*(s)$ to $f_1^*(s)$. Because of the limited state space obtained by grouping similar surface configuration groups, it is possible to solve this dynamic optimization problem (finding the optimal flux profile) using exhaustive enumeration (without eliminating the suboptimal paths at each step) in a short amount of time.

We used eight flux settings (0.06, 0.08, ..., 0.20 ML/s) and found the optimal input profile that would give us minimum deposition time to get to SOM Node 182. Each input profile is a sequence of 10 flux values, and there is a flux value for each coverage interval. Having eight

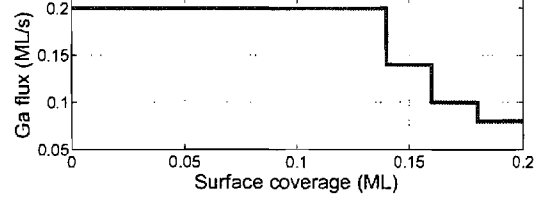


Fig. 12. Optimal flux profile computed by the dynamic model

flux settings and 10 coverage intervals, there are 8^{10} possible input profiles. According to the dynamic model, only 20% of these profiles are able to reach SOM Node 182. Running each simulation takes about 24 hrs utilizing an Intel Xeon processor with a speed of 2.66 GHz, so it would have taken 2.9 million years to run all of the 8^{10} simulations with a single processor. However, using our dynamic model, it took only 5 minutes to predict the evolution of the film structure during these simulations. The minimum cost was obtained with the input profile shown in Figure 12. Figure 13 shows the real and the estimated trajectories of this KMC simulation on SOM3. Again, we have a good agreement between these two trajectories. This particular input profile provided a 48% reduction in the deposition time to reach optimal structure when compared with the constant input KMC simulation under 0.08 ML/s Ga flux. The values of E_{SSC} , $E_{SSC'}$ and $E_{SSC''}$ for the prediction of the final film structure are 0.0159, 0.2207 and 0.2312, respectively. These values are within the range of cumulative distribution functions shown in Figures 10 and 11. In order to visualize the accuracy of this prediction, we also plotted a portion of the SSC functions, which belonged to the actual KMC simulation data, its best matching unit on the map (Node 184) and Node 182 (predicted film structure). According to Figure 14, the simulation data is much noisier than the reconstructions. Also, the lower peak value of the SSC function (around radius=12) of the SOM Node 182 is slightly off compared to the one coming from the KMC simulation. However, the plateau corresponding to the number of step pairs which are distanced with more than 20 lattice units is captured well with the prediction. Hence, these two SSC functions are very similar. This similarity indicates that the dynamic model, once again, does a good job in terms of predicting the final film structure. Furthermore, this prediction can be improved by increasing the size of the training data set and number of SOM nodes in the cell map, or possibly by placing a greater weight on the important, but small sized features, during the training of the SOM.

5 Conclusions

This study describes an approach to understanding and modeling process dynamics from molecular simulations. The proposed algorithm is used on high dimensional Kinetic Monte Carlo (KMC) simulations of epitaxial GaAs thin film deposition. This algorithm consists of three

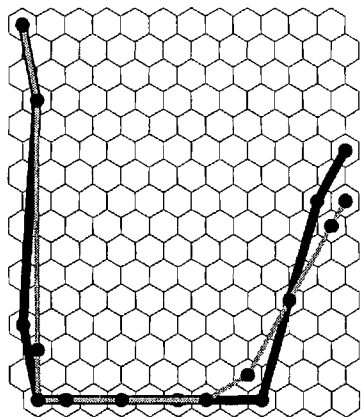


Fig. 13. Trajectories of the KMC simulation (red line) with the optimal flux profile and the prediction (black line) on SOM3

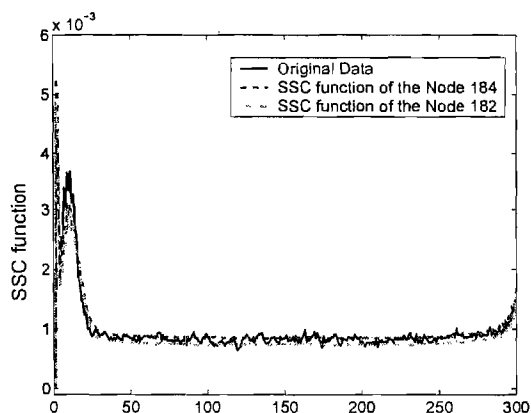


Fig. 14. Reconstructions of the SSC function with the prototype vectors of the SOM nodes 182, 184 and the original simulation data

steps: applying principal component analysis (PCA), self organizing map (SOM) and simple cell mapping (SCM) to identify a dynamic process model. First, a spatial correlation function is used to describe the state space of the system by the characterization of surface snapshots generated under different material flux profiles. Then, a minimum state dimension that can represent the system state is found using PCA. After this reduction step, SOM is employed to group similar surface structures. SOM results led to the identification of a dynamic model through the computation of flux-dependent transitions between surface configuration groups with simple cell mapping (SCM). Analysis of the cell mapping error (CME), which is the one-step prediction error, showed that the structures within the same configuration groups show very similar dynamic behavior under same input conditions. The global error associated with this model reduction approach has also been characterized using

1210 test simulations with highly dynamic input profiles and turned out to be fairly low (less than 0.006 on average for 10890 predictions). Furthermore, the minimization of the deposition time to reach a desired film structure has also been achieved using the compact dynamic model. Our study shows that the proposed algorithm is useful for extracting reduced order dynamic models from high dimensional and noisy molecular simulation data.

Acknowledgements

The authors would like to thank Yannis Kevrekidis, Jay Lee, and Sridhar Ungarala for fruitful discussions. Financial support for this work was provided by the Air Force Office of Scientific Research under award number FA9550-04-1-0183.

References

- [1] P. H. Holloway and G. E. McGuire, *Handbook of compound semiconductors: growth, processing, characterization, and devices*, Noyes Publications, Park Ridge, NJ; 1995.
- [2] M. K. Weldon, K. T. Queeney, J. Eng Jr., K. Raghavachari, and Y. J. Chabal, "The surface science of semiconductor processing: gate oxides in the ever-shrinking transistor," *IEEE Transactions on Industrial Electronics*, vol. 500, no. 1-3, pp. 859-878, 2002.
- [3] A. B. Frazier, R. O. Warrington and C. Friedrich, "The miniaturization technologies: past, present and future," *IEEE Transactions on Industrial Electronics*, vol. 42, no. 5, pp. 423-430, 1995.
- [4] B. Matthes, E. Broszeit, O. Zucker and P. Gauer, "Investigation of thin AlN films piezolayer-field effect transistor applications," *Thin Solid Films*, vol. 226, no. 1, pp. 178-184, 1993.
- [5] I. G. Brown, A. Anders, M. R. Dickinson, R. A. Macgill and O.R. Monteiro, "Recent advances in surface processing with metal plasma and ion beams," *Surface and Coatings Technology*, vol. 112, no. 1-3, pp. 271-277, 1999.
- [6] A. B. Bortz, M. H. Kalos and J. L. Lebowitz, "New algorithm for Monte-Carlo simulation of Ising spin systems," *Journal of Computational Physics*, vol. 17, no. 1, pp. 10-18, 1975.
- [7] P. A. Maksym, "Fast Monte-Carlo simulation of MBE growth," *Semiconductor Science and Technology*, vol. 3, no. 6, pp. 594-596, June 1988.
- [8] M. Itoh, "Atomic-scale homoepitaxial growth simulations of reconstructed III-V surfaces," *Progress in Surface Science*, vol. 66, no. 3-5, pp. 53-153, Feb./Mar. 2001.
- [9] C. W. Gear, I. G. Kevrekidis and C. Theodoropoulos, "Coarse integration/bifurcation analysis via microscopic simulators: micro-Galerkin methods," *Computers and Chemical Engineering*, vol. 26, no. 7-8, pp. 941-963, Aug. 2002.
- [10] A. Bindal, M. G. Ierapetritou, S. Balakrishnan, A. Armaou, A. G. Makeev, I. G. Kevrekidis "Equation-free, coarse-grained computational optimization using timesteppers," *Chemical Engineering Science*, vol. 61, no. 2, pp. 779-793, Jan. 2006.
- [11] D. T. Gillespie, "The chemical Langevin equation," *Journal of Chemical Physics*, vol. 113, no. 1, pp. 297-306, Jul. 2000.

Interpretation of molecular structure and kinetics in melt condensation of A_2 oligomers, B_3 monomers, and monofunctional reagents

C. Oguz, S. Unal, T. E. Long, and M. A. Gallivan

Abstract

This manuscript applies kinetic Monte Carlo simulations to interpret experimental measurements, in the polymerization of hyperbranched poly(ether esters)s in a melt condensation of A_2 oligomers and B_3 monomers. Building on the analytical modeling of Flory and Stockmayer, additional effects of cycle formation, unequal reactivities, and end-capping reagents are added into the simulations, to describe A_2+B_3 polymerization in the absence of a solvent. The experimental data has been published separately¹, and here it is compared to the model predictions in order to quantitatively assess whether the data is consistent with these models. Based on the modeling, we conclude that cycle formation is negligible, suppression of the third B-group is insignificant, and the mobility of the free B_3 monomer leads to enhancement of its reaction rate. The addition of the monofunctional end-capping reagents does not necessarily lead to suppression of branching in the A_2+B_3 system, and depends sensitively on the stoichiometry of the reactants.

1. Introduction

The shape and topology of organic molecules has a profound effect on their properties.² In the last two decades, synthetic polymer chemists have introduced a new class of highly branched macromolecules which are composed of multifunctional monomers, and are classified as either dendrimers or hyperbranched polymers. Dendrimers are typically synthesized using multi-step reactions and they offer superior control of molecular size, shape, and functionality. On the other hand, hyperbranched polymers are less ordered but are easier to synthesize.

Discussion of the synthetic methodologies for preparation of a wide range of hyperbranched and dendritic polymers can be found in several extensive review articles.³⁻⁵ Many of the past experimental studies have focused on the synthesis and characterization of AB_n type monomers ($n \geq 1$), particularly with AB_2 monomers.³⁻⁶ However, very few of the AB_n type monomers are commercially available due to their lack of symmetrical functionality and tendency to react prematurely.⁴ As a result, A_2+B_3 polymerization has recently been the subject of extensive research⁷⁻¹⁴ since it provides an alternative and more convenient way to synthesize highly branched polymers. In contrast to polymerization of AB_n type monomers, these systems offer a wider range of molecular structures depending on the monomeric types and processing conditions. For example, A_2+B_3 polymerization has been performed by heating a mixture of A_2+B_3 ,¹ and also by drop-wise addition of A_2 into B_3 .¹⁵ The molar ratio of A_2 to B_3 can also be varied^{4, 16, 17}. Moreover, our recent efforts have demonstrated the opportunity to control the distance between branch points through the judicious selection of various telechelic oligomers as A_2 .

Modeling studies have long been used to explain experimental observations in an adhoc fashion, and modeling is often used to steer the discovery of synthetic methods and the formation of novel architectures. Early work of Flory¹⁶ and Stockmayer^{18, 19} on the step growth of multifunctional monomers was based on the assumption that there was no cycle formation in these polymerization processes, which enabled the calculation of molecular weight using an infinite series solution. The models of Flory and Stockmayer are useful because they enable quantitative predictions of polymer properties. While it is intuitive that a branched polymer that is composed of A_2 and B_3 monomers will ultimately gel, the model generates the quantitative prediction of gelation at 87% A conversion for the stoichiometry of $A_2:B_3 = 1:1$.¹⁹ More recently, this method has been extended to include the effect of cycle forming reactions on the gel point.²⁰

In order to predict the time-evolution of polymerization, kinetic models based on mass-action kinetics are often employed. Ordinary differential equations are used to describe the concentration of each type of branching unit.^{21, 22} For example, a trifunctional monomer may exist in four states: no reactions (free), one reaction (terminal), two reactions (linear), and fully reacted (dendritic). These models are typically of low dimension and may be solved analytically in some cases, or numerically in others. A disadvantage of this modeling approach is that no information is provided on the molecular weight distribution of the resulting polymer.

When prediction of the molecular weight and its distribution is also desired, population balance models are commonly used, in which the concentration of polymers of each possible size is computed^{17, 21, 23-25}. These models therefore have high dimension. For systems of linear polymers composed of a single monomer type, the

kinetics are fully described by the number of monomers in the polymer. Thus, if one models the concentration of polymers up to a maximum size of 1000 monomeric units, then the dimension of the model is also 1000. However, in branched polymers, more information is needed to describe the kinetics, such as the number of reactive end groups in the polymer. When several descriptors are needed to describe each polymer, the dimension of the population balance model grows rapidly.¹⁷ For the model considered in this manuscript, six descriptors are used to describe each polymer: the number of A_2 monomers, the number of B_3 monomers, the number of unreacted A groups in the polymer, the number and type of unreacted B groups (linear or terminal), and the number of end-capping agents. Solving a population balance model with six descriptors would be very intensive computationally.

Generating functions and the method of moments are often used to reduce the dimension of these population balance models. However, this approach becomes much more difficult as the number of descriptors grows, since moments must be included for each descriptor. The difficulty of this approach is illustrated by the recent paper of Dusek, Duskova-Srmckova, and Voit¹⁷, in which unequal reactivities and monofunctional reagents were considered separately, but not simultaneously.

Cycle formation was neglected by Flory and Stockmayer, but it is a major factor in the structural development of dendritic and hyperbranched polymers^{23, 25-28}. The method of moments has been extended to describe cycle formation in hyperbranched polymers composed of AB_2 monomers²⁵ and the AB_2+B_3 system²⁹. This extension is enabled because in these systems, a maximum of one cycle is possible, at which point the polymer cannot grow further. In contrast, in the A_2+B_3 system, there is no limit on the

maximum number of cycles. Infinite series solutions have been developed to predict the gel point in the A_2+B_3 system, but do not include additional effects such as unequal reactivities of the groups in B_3 , and the effect of monofunctional reagents.²⁰

As an alternative to mass action kinetic models that use concentration variables, Monte Carlo simulations have been performed for hyperbranched polymers, so that more realistic kinetics can be included and so that that structural information can be obtained. For example, the Wiener index can be computed for each polymer in the simulation, which is related to viscosity.³⁰ In the Monte Carlo simulations, individual monomers are reacted with each other to build up the polymer, using random numbers to select each event. Each monomer is tracked throughout the simulation, and information regarding its connection to other monomers is stored. Monte Carlo simulations may be used to describe only the connectivity of the polymers, without describing their spatial positions,^{15, 25, 29, 31} or lattice Monte Carlo simulations can be performed in which each monomer is associated with a spatial position in the lattice.³² Direct comparison between experimental data and Monte Carlo simulations has been limited to date, but these simulations can be valuable in interpreting experimental results.^{15, 25} As computational resources grow, Monte Carlo simulations become an increasingly attractive alternative to population balance modeling, since their major drawback has been the amount of computation required.

Another modeling approach is to describe the structural development of hyperbranched polymers using atomistically detailed algorithms.^{2, 30} These studies are limited to the growth of single polymers at a time because of the high computational cost, so they are not as useful for predicting the molecular weight distribution. However, if the

conformation of the polymer changes throughout the reaction and this strongly influences the kinetics, then it may be necessary to include this level of detail.

Recent studies employ syntheses of hyperbranched polymers via A_2+B_3 both in a solution^{7-9, 12} and also in the melt phase^{1, 33} with no solvent. Herein, we combine experimental and computational efforts to understand the polymer structural development in a melt, especially branching and the onset of gelation. The experimental data has been published previously,¹ and this paper employs a Monte Carlo simulation to interpret the experimental findings and to elucidate the underlying kinetics. In a previous study, we used a simpler kinetic model to explore the role of cyclization in the drop-wise addition of A_2 into B_3 in a solvent.¹⁵ Herein we consider a similar statistical framework to explore a wider range of phenomena, for a batch A_2+B_3 reaction in a melt. The Monte Carlo simulations are used to interpret the experimentally measured number-averaged molecular weight, weight-averaged molecular weight, and the density of branched units, by considering the effects of the cyclization reactions, unequal reactivities, and end-capping on the structure development of hyperbranched polymers.

II. Experimental

Highly branched poly(ether ester)s were synthesized in the melt phase using an oligomeric $A_2 + B_3$ polymerization strategy. Condensation of poly(propylene glycol) (A_2 oligomer) and trimethyl 1,3,5-benzenetricarboxylate (B_3 monomer) generated highly branched structures. The conversion and degree of branching were measured with ^1H NMR spectroscopy, and the molecular weight (number-average, weight-average, and polydispersity) was characterized using size exclusion chromatography (SEC), at six points throughout the polymerization process. Additional experiments were also

performed in which two different monofunctional endcapping reagents were added for the purpose of delaying the gel point. This experimental work has been published previously,¹ and should be consulted for further detail on the experimental procedures. The present work interprets these experiments through a comparison with Monte Carlo simulations, utilizing alternative assumptions and kinetic models. Because the error in the SEC measurements is approximately 10%, our goal is not to achieve exact agreement between the experiments and modeling, but to compare the trends and magnitudes.

III. Model and Simulations

This paper focuses on the use of a model to interpret the experimental data. Each simulation starts with N monomers of A_2 and N monomers of B_3 in the system, since the monomers were polymerized with a 1:1 molar ratio during the experiments. At each step in the simulation, all of the available (unreacted) functional groups are listed. Then, using a random number, an A group and a B group are selected from the list and the reaction is executed. This is followed by updating the list of available A and B groups, molecular weights of the molecules, and the number of dendritic, linear and terminal units in the system.

The probability of selecting a particular pair of A and B groups is proportional to the reaction rate for that pair, which yields the correct time-evolution of the system.³⁴ Therefore, a model is needed for the reaction rates of various events. The first effect that is considered in this effort is the formation of cycles through intramolecular reactions. Our polymerization was performed in the melt, which was hypothesized to minimize cycle formation due to high concentration of reactants, so cyclization reactions are taken into account in the simulations to address this hypothesis. While selecting an A group

(from A₂ oligomers) and a B group (from B₃ monomers) for reaction at each simulation step, some pairs are favored more than the others. If the reaction of an A-B pair leads to cycle formation, the selection probability of that particular pair is promoted by the cyclization parameter γ , such that $\gamma = (k_c/k_{nc})/N$, where k_c is the rate of cyclization reactions for each A-B pair and k_{nc} is the rate of non-cyclization reactions for each A-B pair. N appears in the parameter γ since the number of intermolecular reactions in the simulations grows as N^2 , while intramolecular reactions are initially proportional to N . Although γ is constant throughout each simulation, the overall number of cycle-forming reactions increases with conversion, because the number of possible cycle-forming reactions increases with molecular weight. We use different values of γ in the simulations to explore the effect of cycle formation on the resulting polymer.

In addition to cyclization and non-cyclization reactions, we also consider end-capping reactions between E groups (from monofunctional end-capping reagents) and B groups, with a rate constant of k_e . The ratio $\epsilon = k_e/k_{nc}$ is then the second parameter in our kinetic model. In a second set of simulations, γ and ϵ have been varied to observe their effects on the development of molecular characteristics such as number average molecular weight (M_n), weight average molecular weight (M_w), polydispersity index ($PDI = M_w/M_n$) and the fraction of dendritic units (f_D). f_D is calculated using: $f_D = D / (D+T+L)$, where D , L and T indicate the number of dendritic, linear and terminal units in the system. We plot f_D in the simulation results (as opposed to f_L or f_T), since f_D is the quantity extracted from the ¹H NMR measurements.¹

Depending on the properties of the monomers, such as their molar mass or electrostatic interactions, unequal reactivities of their groups can also affect the structural

development of hyperbranched polymers. End groups usually have higher reactivities than the groups along the length of the chains because of the lower kinetic excluded volume effect.³⁵ This would cause the linear units to have lower reactivities than the terminal units. In order to simulate unequal reactivities of the B groups, we define a third parameter ρ . For each unreacted B group in a B_3 monomer, we check the reaction state of the other two B groups in the monomer. We then consider three possible cases of unequal reactivity. For this purpose, we define k_1 to be the rate of reaction of a B-group in a free B_3 monomer, k_2 to be the rate of a B-group in a terminal unit, and k_3 for a B-group in a linear unit. We assign reaction rates in the ratio of $\rho = k_1/k_2 = k_2/k_3$. k_1 is expected to be enhanced relative to k_2 due to the greater mobility of the free B_3 monomer and its ability to diffuse through the polymer, while k_2 may be different from k_3 due to blocking, free volume, and electrostatic considerations. In order to isolate these two effects, we have also performed simulations with $\rho_{12} = k_1/k_2$ and $k_2 = k_3$, and then also with $\rho_{23} = k_2/k_3$ and $k_1 = k_2$.

For the simulation results presented in this study, the system size is $N=10,000$. Smaller simulation sizes of $N=1000$, 3000, 5000, and 7000 have also been used. The simulation results with $N=1000$ differ significantly from the simulations with larger N . On the other hand, simulations with larger N agreed quantitatively. This suggests that the trends reported in this study are not dependent on the system size. Additionally, for the case of no cycle formation and equal B_3 reactivity, we compared our weight-averaged molecular weight with the analytical theory of Stockmayer,¹⁹ and the error is approximately 1%. Clearly, if one is interested in describing the approach to gelation with no bound on the molecular weight, then the system size would also need to approach

a macroscopic number of monomers (10^{23}), and studies have been performed to quantify this tradeoff.³⁶ However, in these simulations, only molecular weights up to 500,000 g/mol are presented, since this is the range of the experimental data. Furthermore, the error in the SEC measurements is approximately 10%, so an error of 1% in the model predictions is not significant.

III. Results and Discussion

Monte Carlo simulations have been carried out in order to assess the effects of cyclization, unequal reactivities, and end-capping reactions on the polymer structure development. Molecular weights of A_2 (PPG-1000) and B_3 (TMT) are 1060 and 252 g/mol, whereas the molecular weights of two types of end-capping reagents, PPG-M-1000 and dodecanol, are 1200 g/mol and 187 g/mol, respectively. The simulation system containing $10,000 \times 10,000$ A_2 and B_3 monomers yielded molecular weights in the same range as those observed in the experiments. Similar to the experiments, simulations have been initialized with an equal number of A_2 and B_3 monomers in the system. The simulation results plotted are averages over 50 independent realizations.

a. Effects of cyclization reactions

The melt polymerization was previously considered to be sufficiently concentrated so that the cycle formation would be negligible.¹ We first investigate the extent of cyclization via the simulations. Extent of cyclization (EOC) is defined as the fraction of reactions between A and B that is intramolecular, and is a quantity that has been measured previously using MALDI-ToF for linear polymers but is less reliably measured in hyperbranched polymers due to the many possible isomers.^{14, 37}

Figures 1(a) and 1(b) show the experimental¹ and simulated evolution of the weight-average molecular weight M_w and the polydispersity PDI as a function of A_2 conversion, for the 10,000x10,000 A_2+B_3 system with different γ values. The reactivity ratio ρ is 1 and no monofunctional reagents are present. For all γ values, a slow increase in M_w and PDI values was observed until about 80% A_2 conversion. Above 80% A_2 conversion, a sharp increase in M_w and PDI takes place in all the systems, except for the one with the highest level of cyclization ($\gamma = 1$). The experimental data for PDI and M_w are most consistent with a value of γ around 10^{-2} . The ideal limit of no cycle formation, modeled by Flory, is the solid curve with $\gamma = 0$.

M_w and PDI development in systems with cyclization ratios in the range of $\gamma = 0$ to 10^{-2} all agree reasonably well with the experimental data. Due to the variability of the experimental measurements, the goal of the modeling is not to match the experiments exactly, but to assess the magnitude of the various effects. The simulation with $\gamma = 10^{-1}$ has a lower M_w and PDI at high conversions, compared to the experimental data. This suggested that the extent of cyclization, which is the fraction of reactions that are cycle forming, was quite low during the experiment. Figure 2 shows that the system with $\gamma = 10^{-1}$ reaches an extent of cyclization of 0.09 at 90% A_2 conversion. Interestingly, even such a low extent of cyclization dramatically suppressed the M_w and PDI as illustrated in Figures 1(a) and 1(b). Figure 2 also shows that with $\gamma = 10^{-3}$ or $\gamma = 10^{-2}$, the extent of cyclization is less than 3%. This result supports the original hypothesis, that melt polymerization would suppress the effect of cycle formation on molecular weight and gelation.

Another important characteristic of hyperbranched polymers is the fraction of dendritic units f_D , which is directly proportional to the extent of branching in the system. The development of f_D at different γ levels is shown in Figure 1(c) as a function of A_2 conversion. As γ is increased from $\gamma = 10^{-2}$ to $\gamma = 10^{-1}$, an increase in f_D is observed at A_2 conversions of 60% and above. This trend was expected, since cyclization reactions enhance the number of dendritic groups via the formation of small, fully reacted polymers with no free groups. However, the evolution of f_D is not consistent with the experimental data for any value of γ . For all γ , the simulations predict a higher fraction of dendritic units.

b. Effects of unequal reactivities

The disagreement of f_D between the experiments and simulations suggests that there is an additional effect that suppresses the amount of branching during the experiments, other than the effect of cyclization reactions. It could be attributed to the lower reactivity of free B groups in linear units relative to the free B groups in the terminal units or completely unreacted B_3 monomers. However, this would also reduce the molecular weight. In order to assess this trade-off quantitatively, additional simulations have been performed during which the reactivity of free B groups is modified, based on the overall state of the B_3 monomer. In these simulations presented in this section, $\gamma = 0$ since the previous section demonstrated that cyclization did not play a major role in this system.

Figure 3 shows the evolution of the simulations with different levels of reactivity ratio ρ . With $\rho = 1$, the simulations are identical to those in Figure 1, while larger ρ reduces the amount of branching and also the molecular weight and distribution. The reduction in molecular weight was expected since the polymers are becoming more

linear, but the reduction of molecular weight and polydispersity is not as dramatic as it is with cyclization. Even for the extreme value of $\rho = 10$, gelation is delayed but not completely suppressed. In Figure 3(c), the increase in ρ causes a decrease in f_D , as expected, but the shape of the curves does not match the experimental data. In the data, the fraction of dendritic units rises quite high at late conversion, but is very low at earlier conversions. The 1:1 molar ratio of A_2 and B_3 is important in understanding this behavior. Due to the 1:1 molar ratio, there is an excess of B groups, so at full A conversion, only $2/3$ of the B groups have reacted. If the reactivity of the third B group is strongly reduced, then there will be few dendritic units that form, but this is not observed in the experiments. The fact that the experiments eventually reach a large value of f_D near that predicted with $\rho = 1$ instead suggests that a B group in a linear unit has a similar reactivity to a B group in a terminal unit.

The low values of f_D around 60% conversion are more consistent with a suppression of the reaction of terminal units relative to free units, as shown in Figure 4 for various levels of ρ_{12} . Recall that $\rho_{12} = k_1/k_2$, with $k_2 = k_3$. The high reactivity of the free B_3 could be due to its mobility, as well as the fact that B groups in polymers may be partially blocked by other portions of the polymer. This trend in f_D is much more consistent with the experimental data. A high value of $\rho_{12} = 10$ best matches the f_D measurements, while a somewhat lower value of ρ_{12} near 1.5 – 2 agrees best in the molecular weight distribution. Our conclusion based on the simulations in Figures 3 and 4 is that a suppressed reactivity of the third B group in the linear unit is not consistent with the observed data. The more consistent explanation is that the free B_3 monomers are more mobile and therefore react faster than B_3 in polymer.

c. Effects of end-capping reagents

The third and final simulation study considers the addition of monofunctional reagents to the A_2+B_3 system. Stockmayer's theoretical studies of highly branched polymers indicated that addition of a monofunctional end-capping reagent should shift the gel point to higher monomer conversion values.^{16, 18, 19} Thus, delaying the gel point by terminating some of the B functionalities is the main motivation behind using end-capping reagents in this system. A molar ratio of monomers as $A_2:B_3:E = 1:1:1$ was used in the experiments to ensure that residual B end groups, which would be expected in an $A_2:B_3 = 1:1$ system at full A_2 conversion, do not remain in the system at full conversion.

Figure 5 provides a comparison of the change in the evolution of M_w for the addition of PPG-M-1000 (1200 g/mol) at the beginning of the reaction. The A conversion that is plotted also includes the conversion of the end-capping reagent, since that is measured by NMR. Curves are shown for the values of ρ_{12} considered previously in Figure 4. A primary observation is that the addition of the end-capping reagents has the larger effect when the reactivity ratio is also large. Furthermore, this effect is only observed when the end-capping reagents also have a higher reactivity than the A groups in A_2 ($\epsilon \gg 1$). This might be the case for the dodecanol reagent, if the end-capping reagent has a higher diffusivity than the A_2 due to its lower molar mass.

In the simulations presented in Figures 5 and 6, we set $\epsilon = 1000$, although the results are similar for $\epsilon = 10$. When $\epsilon \approx 1$, the simulations predict that the end-capping reagents have a negligible effect on the polymer structure. By reacting with the excess B groups, they only add their extra mass to the polymer. In the opposite limit, when $\epsilon \gg 1$ and ρ_{12}

$\gg 1$, the E groups react quickly with the free B_3 monomers, after which the EB_3 units begin reacting with A_2 . Each B_3 is thus bonded to only two A_2 monomers, so the polymers have a linear structure.

In the experiments, it was observed that the gel point was completely suppressed up to 98% conversion of each monomer, and the measured g' contraction factor from GPC was more consistent with a highly branched polymer (large f_D). Our simulations do predict a suppression of molecular weight with the end-capping reagents, but gelation is only delayed and not completely suppressed. At the extreme value $\rho_{12} = 10$, a significant reduction in f_D is also implied, but at lower values of ρ_{12} , high levels of branching are still predicted.

In the presentation of the experimental results,¹ we suggested that ester interchange of the A_2 with the monofunctional reagents might account for the observed reduction in molecular weight. While this interchange would cause a randomization of the polymer, at the high conversion of 98% it does not provide a consistent explanation for the extreme reduction in molecular weight. At 98% conversion, most of the A_2 monomers freed up by ester interchange would have reacted with another B_3 monomer.

Other effects not included in the model could be causing the suppression of gelation observed in the experiments, such as the spatial distribution of the monomers in the polymer. These effects could be exacerbated when the end-capping reagents are added, since all B groups must eventually react, even those buried or blocked in the center of the spherical polymer. Possibly, at high conversion, such groups are more likely to undergo cyclization reactions, which would suppress the molecular weight.

An alternative explanation was also suggested by our simulations. We observed that the simulation results are extremely sensitive to the stoichiometry near $A_2:B_3:E = 1:1:1$. In particular, if there is a reduction in the amount of B_3 , then not all of the A_2 groups will be able to react with B groups, and the molecular weight will be reduced. This may be a particular issue in our experiments, since B_3 loss may be facilitated by the nitrogen purge at our final polymerization temperature of 180 °C. Due to the uncertainty in our final stoichiometry measurements it is not possible to eliminate this effect. The simulations suggest that that our chosen stoichiometry of 1:1:1 is not a robust operating point, due to the extreme sensitivity of the molecular weight on the stoichiometry. Figure 6 shows a comparison of the simulations with $A_2:B_3:E = 1:1:1$ and with 1:0.9:1. At this stoichiometry, 90% A_2 conversion is the maximum that can be achieved, and gelation is completely suppressed at full B_3 conversion.

4. Conclusions

The formation of highly branched poly (ether ester)s by the melt condensation of an A_2 oligomer with a B_3 monomer has been studied using experiments and kinetic Monte Carlo simulations. The simulations demonstrated that unequal reactivities can play an important role in the structure development of hyperbranched polymers, even when it has a little impact on the molecular weight. The results also indicate that the presence of end-capping reagents delays the gel point. However, the effect of end-capping agents also depends strongly on the ratios of the various monomers and their reactivity ratios. These results are motivating our further study of the role of end-capping reagents in the $A_2 + B_3$ system. In summary, the kinetic Monte Carlo simulations provide a tool for quantitatively assessing the effects of simple reaction mechanisms on molecular

structure evolution, enabling the consideration of a broader range of mechanisms than with analytical models.

Acknowledgments

This material is based upon work supported by, or in part by, the U. S. Army Research Laboratory and the U. S. Army Research Office under contract/grant number DAAD 19-02-1-0275 Macromolecular Architecture for Performance (MAP) MURI, and the Air Force Office of Scientific Research under award number FA9550-04-1-1083. The authors would also like to thank Eastman Chemical Company for financial support and acknowledge Iskender Yilgor and Emel Yilgor for their role in initiating this collaboration.

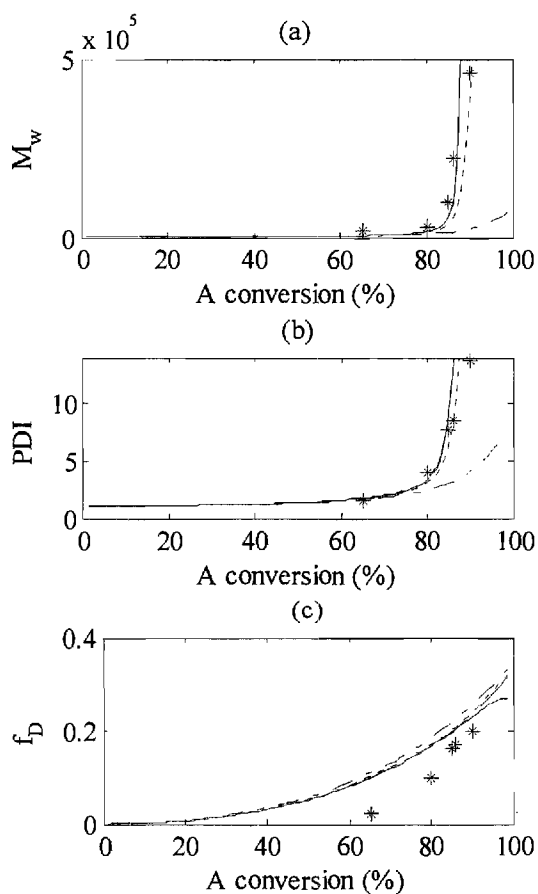


Figure 1. Comparison of simulation and experiment¹ (*). In the simulations, the cyclization ratio γ is varied: $\gamma = 0$ (solid line), $\gamma = 10^{-3}$ (dashed line), $\gamma = 10^{-2}$ (dotted line), $\gamma = 10^{-1}$ (dashed-dot line). (a) Weight-average molecular weight M_w (b) Polydispersity index PDI (c) Fraction of dendritic units f_D . Agreement between experiments and simulations is not achieved for f_D at any value of γ .

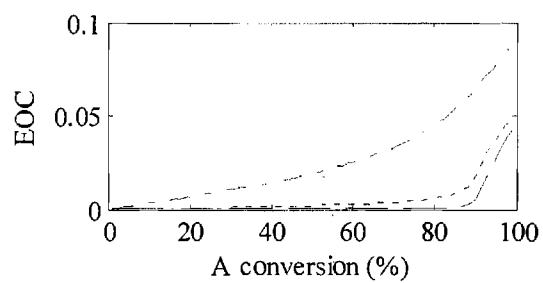


Figure 2. Simulation predictions of extent of cyclization. The cyclization ratio γ is varied: $\gamma = 10^{-3}$ (dashed line), $\gamma = 10^{-2}$ (dotted line), $\gamma = 10^{-1}$ (dashed-dot line). For the $\gamma = 0$ case, EOC = 0.

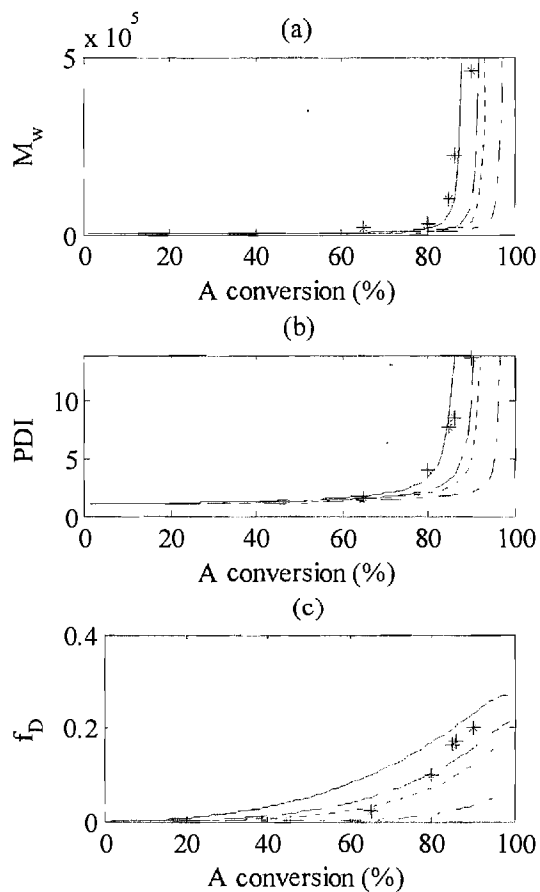


Figure 3. Comparison of simulation and experiment¹ (*). In the simulations, $\rho = k_1/k_2 = k_2/k_3$. $\rho = 1$ (solid line), $\rho = 1.5$ (dashed line), $\rho = 2$ (dotted line), $\rho = 10$ (dashed-dot line). (a) Weight-averaged molecular weight (b) polydispersity index (c) fraction of dendritic units.

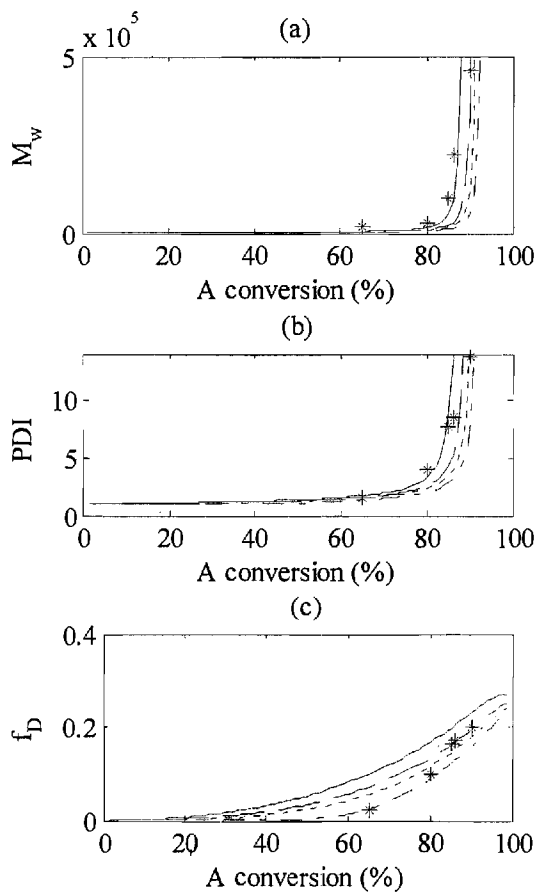


Figure 4. Comparison of simulation and experiment¹ (*). In the simulations, $\rho_{12} = k_1/k_2$ and $k_2 = k_3$. $\rho_{12} = 1$ (solid line), $\rho_{12} = 1.5$ (dashed line), $\rho_{12} = 2$ (dotted line), $\rho_{12} = 10$ (dashed-dot line). (a) Weight-averaged molecular weight (b) polydispersity index (c) fraction of dendritic units.

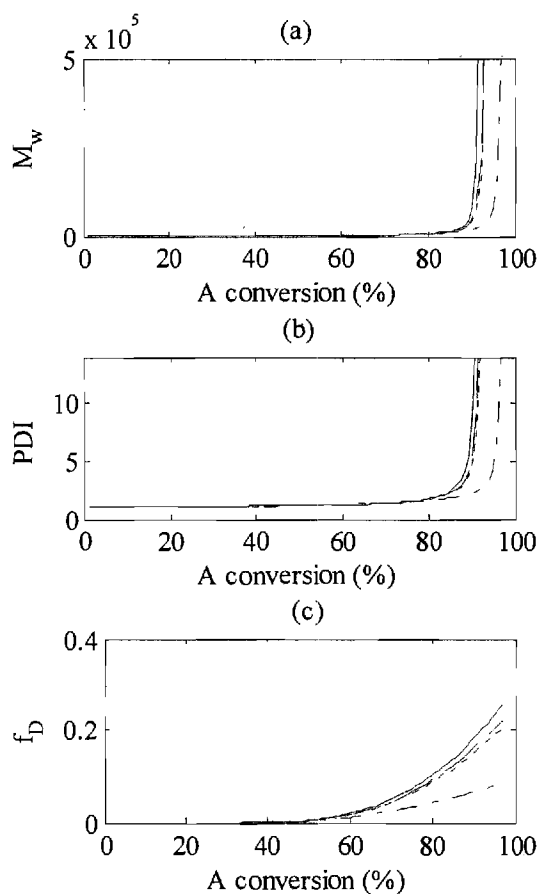


Figure 5. Simulated evolution with end-capping reagents added at the beginning of the process, with $A_2:B_3:E = 1:1:1$. The molecular weight of E (PPG-M-1000) is $M_{W, \text{cap}} = 1200 \text{ g/mol}$. As in Figure 4, $\rho_{12} = k_1/k_2$ and $k_2 = k_3$. $\rho_{12} = 1$ (solid line), $\rho_{12} = 1.5$ (dashed line), $\rho_{12} = 2$ (dotted line), $\rho_{12} = 10$ (dashed-dot line). (a) Weight-averaged molecular weight (b) polydispersity index (c) fraction of dendritic units. (Note: Dendritic units are calculated here based on the number of A-B reactions. E-B reactions are not considered in the calculation since they do not lead to further branching.)

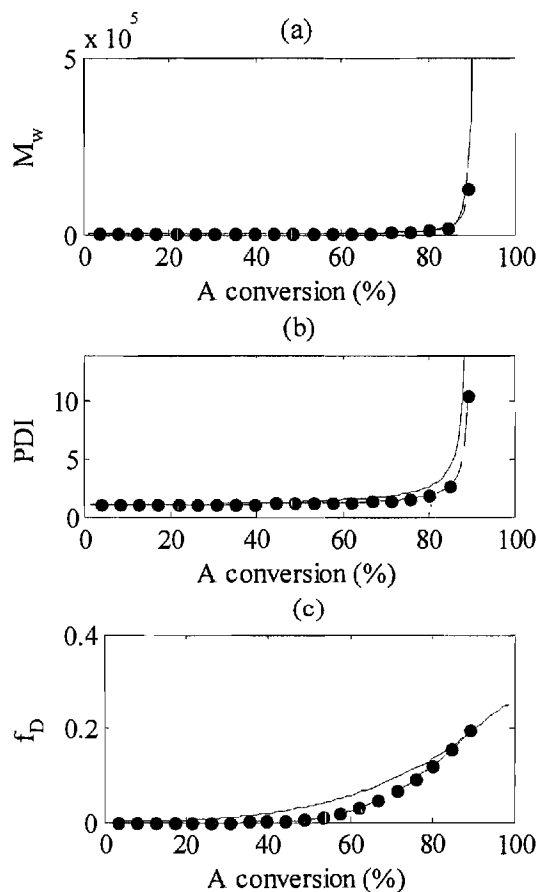


Figure 6. Simulated evolution with end-capping reagents added at the beginning of the process, with variation in stoichiometry: $A_2:B_3:E = 1:1:1$ (solid line), $A_2:B_3:E = 1:0.9:1$ (dotted line with markers). The molecular weight of E (PPG-M-1000) is $M_{w,cap} = 1200$ g/mol. $\rho_{12} = k_1/k_2 = 1.5$ and $k_2 = k_3$. (a) Weight-averaged molecular weight (b) polydispersity index (c) fraction of dendric units.

References

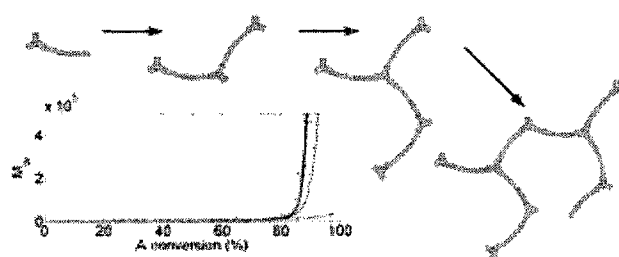
1. Unal, S.; Long, T. E. *Macromolecules* **2006**, 39, 2788.
2. Aerts, J. *Computational and Theoretical Polymer Science* **1998**, 8, (1/2), 49.
3. Kim, Y. H. *Journal of Polymer Science: Part A: Polymer Chemistry* **1998**, 36, 1685.
4. Voit, B. *Journal of Polymer Science: Part A: Polymer Chemistry* **2005**, 43, 2679.
5. Jikei, M.; Kakimoto, M. *Progress in Polymer Science* **2001**, 26, 1233.
6. Gao, C.; Yan, D. *Progress in Polymer Science* **2004**, 29, 183.
7. Jikei, M.; Chon, S. H.; Kakimoto, M. A.; Kavauchi, S.; Imase, T.; Watanabe, J. *Macromolecules* **1999**, 32, 2061.
8. Kricheldorf, H. R.; Vakhtangishvili, L.; Fritsch, D. J. *Journal of Polymer Science: Part A: Polymer Chemistry* **2002**, 40, (17), 2967.
9. Fang, J.; Kita, H.; Okamoto, K. *Macromolecules* **2000**, 33, 4639.
10. Emrick, T.; Chang, H. T.; Frechet, J. M. J. *Macromolecules* **1999**, 32, 6380.
11. Komber, H.; Voit, B. I.; Monticelli, O.; Russo, S. *Macromolecules* **2000**, 34, 5487.
12. Lin, Q.; Long, T. E. *Macromolecules* **2003**, 36, 9809.
13. Yan, D.; Gao, C. *Macromolecules* **2000**, 33, 7693.
14. Kricheldorf, H. R.; Fritsch, D.; Vakhtangishvili, L.; Schwarz, G. *Macromolecules* **2003**, 36, 4337.
15. Unal, S.; Oguz, C.; Yilgor, E.; Gallivan, M.; Long, T. E.; Yilgor, I. *Polymer* **2005**, 46, 4533.
16. Flory, P. J. *Chemical Reviews* **1946**, 39, (1), 137.
17. Dusek, K.; Duskova-Smrckova, M.; Voit, B. *Polymer* **2005**, 46, 4265.
18. Stockmayer, W. H. *Journal of Chemical Physics* **1943**, 11, (2), 45.
19. Stockmayer, W. H. *Journal of Polymer Science* **1952**, 9, (1), 69.
20. Cail, J. I.; Stepto, R. F. T. *Polymer Bulletin* **2007**, 58, 15.
21. Radke, W.; Litwinenko, G.; Muller, A. H. E. *Macromolecules* **1998**, 31, 239.
22. Holter, D.; Frey, H. *Acta Polymerica* **1997**, 48, 298.
23. Hanselmann, R.; Holter, D.; Frey, H. *Macromolecules* **1998**, 31, 3790.
24. Zhou, Z.; Yan, D. *Polymer* **2006**, 47, 1473.
25. Dusek, K.; Somvarky, J.; Smrckova, M.; Simonsick, W. J.; Wilczek, L. *Polymer Bulletin* **1999**, 42, 489.
26. Kricheldorf, H. R.; Vakhtangishvili, L.; Fritsch, D. J. *Journal of Polymer Science: Part A: Polymer Chemistry* **2002**, 40, (17), 2967-2978.
27. Gong, C.; Miravet, J.; Frechet, J. M. J. *Journal of Polymer Science: Part A: Polymer Chemistry* **1999**, 37, (16), 3193.
28. Kricheldorf, H. R.; Schwarz, G. *Macromolecular Rapid Communications* **2003**, 24, 359.
29. Galina, H.; Lechowicz, J. B. *e-Polymers* **2002**, 12, 1.
30. Widmann, A. H.; Davies, G. R. *Computational and Theoretical Polymer Science* **1998**, 8, (1/2), 191.
31. Somvarky, J.; Dusek, K. *Polymer Bulletin* **1994**, 33, 369.
32. Cameron, C. *Journal of Chemical Physics* **1998**, 108, (19), 8235.
33. Stumbe, J.; Bruchmann, B. *Macromolecular Rapid Communications* **25**, 921.

34. Bortz, A. B.; Kalos, M. H.; Lebowitz, J. L. *Journal of Computational Physics* **1975**, 17, (1), 10.
35. McKee, M. G.; Unal, S.; Wilkes, G. L.; Long, T. E. *Progress in Polymer Science* **2005**, 30, 507.
36. Somvarsky, J.; Dusek, K. *Polymer Bulletin* **1994**, 33, 377.
37. Kricheldorf, H. R.; Lomadze, N.; Polefka, C.; Schwarz, G. *Macromolecules* **2006**, 39, 2107.

For Table of Contents Use Only

**Interpretation of molecular structure and kinetics in melt condensation of A_2
oligomers, B_3 monomers, and monofunctional reagents**

C. Oguz, S. Unal, T. E. Long, and M. A. Gallivan



The TOC graphic is adapted from a figure in this paper.

The TOC graphic should appear in color.