Path Factorization Approach to Stochastic Simulations

Manuel Athènes[1] and Vasily V. Bulatov[2]
[1]CEA, DEN, Service de Recherches de Métallurgie Physique, F-91191 Gif-sur-Yvette, France
[2]Lawrence Livermore National Laboratory, Livermore, California 94551, USA
(Received 24 March 2014; published 3 December 2014)

The computational efficiency of stochastic simulation algorithms is notoriously limited by the kinetic trapping of the simulated trajectories within low energy basins. Here we present a new method that overcomes kinetic trapping while still preserving exact statistics of escape paths from the trapping basins. The method is based on path factorization of the evolution operator and requires no prior knowledge of the underlying energy landscape. The efficiency of the new method is demonstrated in simulations of anomalous diffusion and phase separation in a binary alloy, two stochastic models presenting severe kinetic trapping.

DOI: 10.1103/PhysRevLett.113.230601 PACS numbers: 05.10.Ln, 05.20.Jj, 07.05.Tp

The time evolution of many natural and engineering systems is described by a master equation (ME), i.e., a set of ordinary differential equations for the time-dependent vector of state probabilities [1,2]. For models with a large (or infinite but countable) number of states, direct solution of the ME is prohibitive and a kinetic Monte Carlo (KMC) algorithm is used instead to simulate the time evolution by generating sequences of stochastic transitions from one state to the next [3–5]. Statistically equivalent to the (most often unknown) solution of the ME, KMC calculations find a growing number of applications in natural and engineering sciences. However still wider applicability of KMC simulations is severely limited by the notorious kinetic trapping where the stochastic trajectory repeatedly visits a subset of states, a trapping basin, connected to each other by high-rate transitions while transitions out of the trapping basin are infrequent and take a great many of KMC steps to observe.

In this Letter, we present an efficient method for sampling stochastic trajectories escaping from the trapping basins. Unlike recent methods that focus on short portions of the full kinetic path directly leading to the escapes and/or require equilibration over a path ensemble [6–12], our method constructs an entire stochastic trajectory within the trapping basin, including the typically large numbers of repeated visits to each trapping state as well as the eventual escape. Referred hereafter as kinetic path sampling (KPS), the new algorithm is statistically equivalent to the standard KMC algorithm and entails (i) iterative factorization of paths inside a trapping basin, (ii) sampling a single exit state within the basin's perimeter, and (iii) generating a first-passage path and an exit time to the selected perimeter state through an exact randomization procedure. We demonstrate the accuracy and efficiency of the KPS algorithm on two models: (1) diffusion on a random energy landscape specifically designed to yield a wide and continuous spectrum of time scales and (2) kinetics of phase separation in supersaturated solid solutions of copper in iron. The proposed method is immune to kinetic trapping and performs well under physical conditions where the standard KMC simulations slow down to a crawl. In particular, it reaches later stages of phase separation in the Fe-Cu system and captures a qualitatively new kinetics and mechanism of copper precipitation.

The evolution operator, obtained formally from solutions of the ME, can be expressed as an exponential of the time-independent transition rate matrix [13]

$$\mathbf{P}(t, t+\tau) = \exp\left(\int_t^{t+\tau} \mathbf{M} ds\right) = \exp(\tau \mathbf{M}), \quad (1)$$

where $P_{\beta\gamma}(t, t+\tau)$ is the probability to find the system in state γ at $t+\tau$ given that it was in state β at time t, $M_{\beta\gamma}$ is the rate of transitions from state β to state γ (off-diagonal elements only), and the standard convention is used to define the diagonal elements as $M_{\beta\beta} = -\sum_{\nu\neq\beta} M_{\beta\nu}$. As defined, the evolution operator belongs to the class of stochastic matrices such that $\sum_\nu P_{\beta\nu} = 1$ and $P_{\beta\gamma} \geq 0$ for any β, γ, t, and τ. If known, the evolution operator can be used to sample transitions between any two states and over arbitrary time intervals τ [14]. In particular, substantial simulation speed-ups can be achieved by sampling transitions to distant states on an absorbing perimeter of a trapping basin. Two main deficiencies of the existing implementations of this idea [15–18] are that states within the trapping basin are expected to be known *a priori* and that computing the evolution operator requires a partial eigenvalue decomposition of $\mathbf{M}$ entailing high computational cost [19]. In contrast, the KPS algorithm does not require any advance knowledge of the trapping basin nor does it entail matrix diagonalization. Instead, KPS detects kinetic trapping and charts the trapping basin iteratively, state by state, and achieves high computational efficiency by sequentially eliminating all the trapping states through

path factorization [20–22]. Here, Wales's formulation [22] of path factorization is adopted for its clarity.

Consider the linearized evolution operator

$$\mathbf{P}^{(0)}(t, t+\tau) = \mathbf{I} + \tau\mathbf{M}, \tag{2}$$

where $\mathbf{I} = \mathbf{P}^{(0)}(t,t)$ is the identity matrix. Assuming that $\tau \leq \min\{-(M_{\beta\beta})^{-1} : \forall\beta\}$, $\mathbf{P}^{(0)}$ is a proper stochastic matrix that can be used to generate stochastic sequences of states from the ensemble of paths defined by matrix $\mathbf{M}$. The diagonal elements of $\mathbf{P}^{(0)}$ define the probabilities of round-trip transitions after which the system remains in the same state. To correct for the linearization of the evolution operator in Eq. (2), the time elapsed before any transition takes place is regarded as a stochastic variable and is sampled from an exponential distribution with density $t \to \tau^{-1}\exp(-t/\tau)$ [23]. This simple time randomization obviates the need for exponentiating the transition rate matrix in Eq. (1). Following Ref. [22], consider a bidirectional connectivity graph defined by $\mathbf{P}^{(0)}$ in which N states in the trapping basin are numbered in order of their elimination, $\mathbb{E} = \{1, 2, \ldots, N\}$. An iterative path factorization procedure then constructs a set of stochastic matrices $\{\mathbf{P}^{(n)}\}_{0\leq n\leq N}$ such that, after the nth iteration, all states $\gamma \leq n$ are eliminated in the sense that the probability of a transition from any state β to state $\gamma \leq n$ is zero. Specifically, at the n step of factorization the transition probability $P^{(n)}_{\beta\gamma}$ ($\gamma > n$, $\forall\beta$) is computed as the sum of the probability of a direct transition $P^{(n-1)}_{\beta\gamma}$ and the probabilities of all possible indirect paths involving round-trips in n after having initially transitioned from β to n and before finally transitioning from n to γ, e.g., $\beta \to \gamma$, $\beta \to n \to \gamma$, $\beta \to n \to n \to \gamma$, $\beta \to n \to n \to n \to \gamma$, and so on. With the round-trip probability being $P^{(n-1)}_{nn}$, it is a simple matter to sum the geometric series corresponding to the round-trip paths. Although any intermediate $\mathbf{P}^{(n)}$ can be used to generate stochastic escapes from any state $\alpha \in \mathbb{E}$, a trajectory generated using $\mathbf{P}^{(N)}$ is the simplest containing a single transition from α that effectively subsumes all possible transitions involving the deleted states in the trapping basin $\mathbb{E}$. On the other end, a detailed escape trajectory can be generated using $\mathbf{P}^{(0)}$ that accounts for all transitions within $\mathbb{E}$, reverting back to the standard KMC simulation. Remarkably, it is possible to efficiently construct statistically correct escape trajectories without ever performing any detailed (and inefficient) KMC simulation.

Consider matrix $\mathbf{H}^{(n)}$ whose elements $H^{(n)}_{\beta\gamma}$ store the number of $\beta \to \gamma$ transitions observed in a stochastic simulation with $\mathbf{P}^{(n)}$. Given $\mathbf{H}^{(n)}$, one can randomly generate $\mathbf{H}^{(n-1)}$, the matrix similarly used to count the transition numbers observed in a stochastic process based on $\mathbf{P}^{(n-1)}$ without actually performing the simulation using $\mathbf{P}^{(n-1)}$. The ratio of transition probabilities ($\gamma > n$)

$$R^{(n)}_{\beta\gamma} = P^{(n-1)}_{\beta\gamma}/P^{(n)}_{\beta\gamma} \tag{3}$$

defines the conditional probability that a trajectory generated using $\mathbf{P}^{(n-1)}$ contains a direct transition from β to γ given that the trajectory generated with $\mathbf{P}^{(n)}$ contains the same transition. For $\beta = n$, $R^{(n)}_{n\gamma}$ is independent of γ and is equal to $1 - P^{(n-1)}_{nn}$, the probability of escape from n. It is thus possible to generate $\mathbf{H}^{(n-1)}$ by performing a stochastic simulation with $\mathbf{P}^{(n)}$, harvesting $\mathbf{H}^{(n)}$, and drawing random variates from (standard and negative) binomial distributions whose exponents and coefficients are given by the elements of $\mathbf{H}^{(n)}$ and $\mathbf{R}^{(n)}$, respectively. This randomization procedure can be used iteratively on n in the reverse order from N to 1 to generate $\mathbf{H}^{(0)}$ containing a detailed count of transitions involving all states in $\mathbb{E}$. Finally, the time of exit out of $\mathbb{E}$ is sampled by drawing a random variate from the gamma distribution whose scale and shape parameters are defined by τ and the total number of transitions contained in $\mathbf{H}^{(0)}$, respectively. Thus, in its simplest form the KPS algorithm proceeds by first deleting all states in $\mathbb{E}$ through iterative forward path factorization, then using $\mathbf{P}^{(N)}$ to sample a single transition from $\alpha \in \mathbb{E}$ and to generate $\mathbf{H}^{(N)}$, followed by a backward randomization to reconstruct a detailed stochastic path within $\mathbb{E}$ and to sample an escape time to the selected exit state. A detailed description of the KPS algorithm is given in the Supplemental Material [24].

We first apply KPS to simulations of a random walker on a disordered energy landscape (substrate) [25]. The substrate is a periodically replicated 256×256 fragment of the square lattice on which the walker hops to its four nearest-neighbor (NN) sites with transition rates

$$M_{\beta\gamma} = \exp[(E_\beta - E^s_{\beta\gamma})/T],$$

where T is the temperature, E_β the site energy, and $E^s_{\beta\gamma}$ the saddle energy between sites β and γ. The energy landscape is purposefully constructed to contain trapping basins of widely distributed sizes and depths (see the Supplemental Material [24] for details) and is centered around the walker's initial position next to the lowest energy saddle (Fig. 1).

When performed at temperature $T = 2.5$, standard KMC simulations (with hops only to the NN sites) are efficient, enabling the walker to explore the entire substrate. However at $T = 1$, the walker remains trapped near its initial position, repeatedly visiting states within a trapping basin. To chart a basin set $\mathbb{E}$ for subsequent KPS simulations, the initial state 1 is eliminated at the very first iteration followed in sequence by the "most absorbing states" for which $P^{(n)}_{1\gamma}$ is found to be largest at the nth iteration ($2 \leq n \leq N$). The expanding contours shown in Fig. 1 depict the absorbing boundary $\partial\mathbb{A}$ (perimeter of the basin) obtained after eliminating 2^7, 2^9, 2^{11}, 2^{12}, 2^{13}, and 2^{14} states. The perimeter contour $\partial\mathbb{A}$ consists of all states γ for which $P^{(N)}_{1\gamma}$ is nonzero.

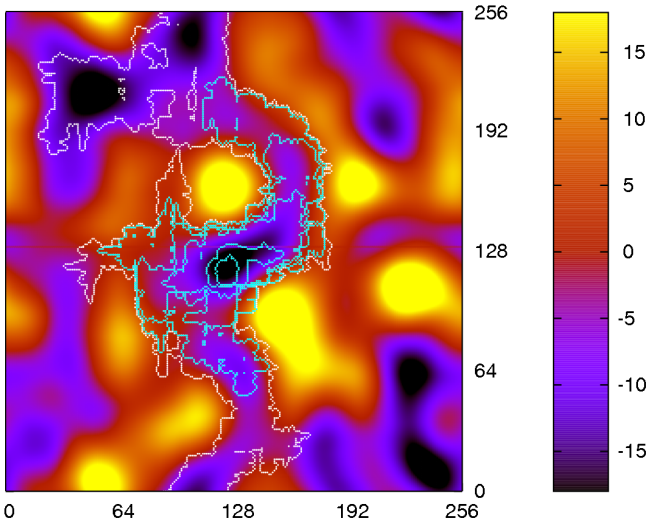

FIG. 1 (color online). Energy surface of the saddle points. The color scale is in the units of ϵ. Artificial smoothing is used for better visualization.

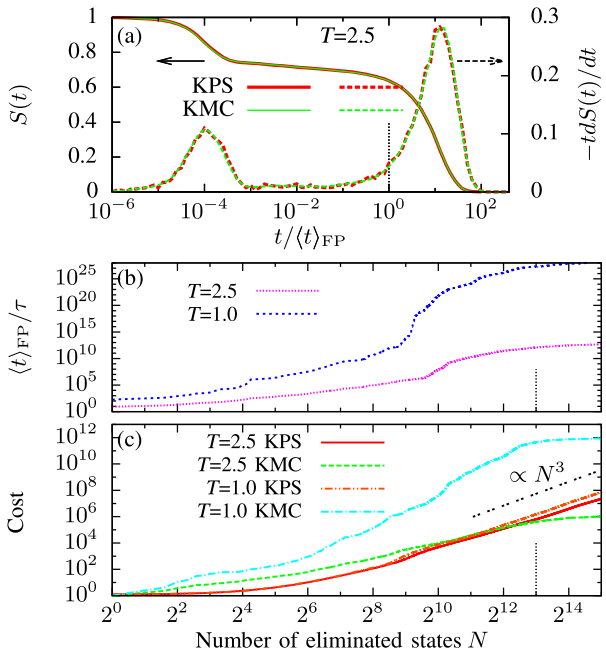

FIG. 2 (color online). (a) The probability $S(t)$ for the walker to remain within a trapping basin containing $N = 2^{13}$ states (solid line) and the distribution of times of escape out of the same basin (dashed lines) using a log scale for the bins. (b) The mean first-passage time $\langle t \rangle_{\mathrm{FP}}$ as a function of the number of states N included in the trapping basin. (c) Computational cost of KMC and KPS simulations as a function of N at two different temperatures (in the units of a single KMC hop).

To demonstrate correctness of KPS, we generated 10^4 paths starting from state (127,127) and ending at the absorbing boundary $\partial\mathbb{A}$ of the basin containing $N = 2^{13}$ states, performing both KPS and KMC simulations at $T = 2.5$. The perfect match between the two estimated distributions of exit times is shown in Fig. 2(a). The mean times of exit to $\partial\mathbb{A}$ are plotted as a function of the number of eliminated states at $T = 2.5$ and $T = 1.0$ in Fig. 2(b), while the costs of both methods are compared in Fig. 2(c). At $T = 1.0$, KMC trajectories are trapped and never reach $\partial\mathbb{A}$: in this case, we plot the expectation value for the number of KMC hops required to exit $\mathbb{E}$, which is always available after path factorization [22]. We observe that the KPS cost scales as N^3, as expected for this factorization, and exceeds that of KMC for $N > 2^{12}$ at $T = 2.5$. However, at $T = 1$, trapping becomes severe, rendering the standard KMC algorithm inefficient and the wall clock speed-up achieved by KPS is 4 orders of magnitude for $N = 2^{15}$. We observe that with the KPS algorithm the net cost of generating an exit trajectory is nearly independent of the temperature but grows exponentially with the decreasing temperature in KMC simulations. At the same time, an accurate measure of the relative efficiency of KPS and KMC simulations is always available in path factorization, allowing one to revert to the standard KMC algorithm whenever it is relatively more efficient. Thus, when performed correctly, a stochastic simulation combining KPS and KMC algorithms should always be more efficient than one resorting to KMC algorithm alone.

As a second illustration, we apply KPS to simulate the kinetics of copper precipitation in iron within a lattice model parametrized using electronic structure calculations [26]. The simulation volume is a periodically replicated fragment of the body-centered-cubic lattice with $128 \times 128 \times 128$ sites on which 28 163 Cu atoms are initially randomly dispersed. Fe atoms occupy all of the remaining lattice sites except one that is left vacant allowing atom diffusion to occur by vacancy (V) hopping to one of its NN lattice sites. Its formation energy being substantially lower in Cu than in Fe, the vacancy is readily trapped in Cu precipitates rendering the KMC algorithm grossly inefficient below 550 K [26]. Whenever the vacancy is observed to attach to a Cu cluster, we perform KPS over a precharted set $\mathbb{E}$ containing N trapping states that correspond to all possible vacancy positions inside the V-Cu_{N-1} cluster containing $N - 1$ Cu atoms: the shape of the trapping cluster is fixed at the instant when the vacancy first attaches. The fully factored matrix $\mathbf{P}^{(N)}$ is then used to propagate the vacancy to a lattice site just outside the fixed cluster shape, which is often followed by vacancy returning to the same cluster. If the newly formed trapping cluster has the same shape as before, the factorized matrix is used again to sample yet another escape. However, a new path factorization (KPS cycle) is performed whenever the vacancy reattaches to the same Cu cluster but in a different cluster shape or attaches to another Cu cluster (see the Supplemental Material for additional simulation details [24]).

We simulated copper precipitation in iron at three different temperatures, $T_0 = 273$ K, $T_1 = 373$ K, and $T_2 = 473$ K, for which the atomic fraction of Cu atoms

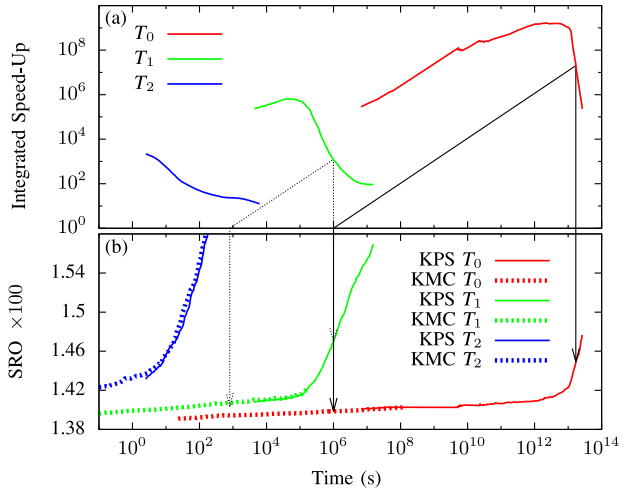

FIG. 3 (color online). (a) Integrated speed-up plotted as a function of the physical time simulated by KPS. (b) Time evolution of the averaged SRO in KPS and KMC simulations at three different temperatures.

used in our simulations significantly exceeds copper solubility limits in iron. Defined as the ratio of physical time simulated by KPS to that reached in KMC simulations over the same wall clock time, the integrated speed-up is plotted in Fig. 3(a) as a function of the physical time simulated by KPS (averaged over 41 simulations for each method and at each temperature).

The precipitation kinetics are monitored through the evolution of the volume-averaged Warren-Cowley short-range order (SRO) parameter [27] shown in Fig. 3(b) for both KPS and KMC simulations. At T_0 and T_1 the kinetics proceed through a distinct incubation stage reminiscent of a time lag associated with repeated redissolution of subcritical nuclei prior to reaching the critical size in the classical nucleation theory [28]. However, "incubation" observed here is of a distinctly different nature since all of our simulated solid solutions are thermodynamically unstable, and even the smallest of Cu clusters, once formed, never dissolve. At all three temperatures the growth of V-Cu_{N-1} clusters is observed to proceed not through the attachment of mobile V-Cu dimers but primarily through the cluster's own diffusion and sweeping of neighboring immobile Cu monomers [24]. This is consistent with an earlier study that also suggested that, rather counterintuitively, the diffusivity of V-Cu_{N-1} clusters should increase with the increasing N before tapering off at $N = 30$–100 (see Fig. 9 of Ref. [26]). We further observe that at T_0 the crossover from the slow initial incubation to faster "agglomeration" growth seen in Fig. 3(b) occurs concomittantly with the largest cluster growing to 15—16 Cu atoms [24]. Individual realizations of the stochastic precipitation kinetics reveal that, in addition to $N = 15$, cluster growth slows down again once the cluster reaches $N = 23$, 27, 35, and so on (see Fig. S4 in the Supplemental Material [24]). Leaving precise characterization of these transitions to future work, we speculate that the observed "magic numbers" correspond to compact clusters with fully filled NN shells in which vacancy trapping is particularly strong, reducing the rate of shape-modifying vacancy escapes required for cluster diffusion.

Numerically, as expected, the integrated speed-up rapidly increases with the decreasing temperature as vacancy trapping becomes more severe. Two line segments of unit slope and two pairs of vertical arrows are drawn in Fig. 3 to compare evolution stages achievable within KPS and KMC simulations over the same wall clock time. As marked by the pair of two solid vertical arrows on the right, the integrated speed-up exceeds 7 orders of magnitude at T_0. Subsequent reduction in the speed-up coincides with the transition into the agglomeration regime where increasingly large V-Cu clusters repeatedly visit an increasingly large number of distinct shapes [29]. Unquestionably, the efficiency of KPS simulations for this particular model can be improved by indexing distinct cluster shapes for each cluster size and storing the path factorizations to allow for their repeated use during the simulations [30]. In any case, given its built-in awareness of the relative cost measured in KMC hops, KPS is certain to enable more efficient simulations of diffusive phase transformations in various technologically important materials. In particular, it is tempting to relate an anomalously long incubation stage observed in aluminium alloys with Mg, Si, and Se additions [31] to possible trapping of vacancies on Se, similar to the retarding effect of Cu on the aging kinetics reported here for the Fe-Cu alloys.

In summary, we developed a kinetic path sampling algorithm suitable for simulating the evolution of systems prone to kinetic trapping. Unlike most other algorithms dealing with this numerical bottleneck [15–17,30,32,33], KPS does not require any *a priori* knowledge of the properties of the trapping basin. It relies on an iterative path factorization of the evolution operator to chart possible escapes, measures its own relative cost, and reverts to the standard KMC algorithm if the added efficiency no longer offsets its computational overhead. At the same time, the KPS algorithm is exact and samples stochastic trajectories from the same statistical ensemble as the standard KMC algorithms. Being immune to kinetic trapping, KPS is well positioned to extend the range of applicability of stochastic simulations beyond their current limits. Furthermore, KPS can be combined with spatial protection [34] and synchronous or asynchronous algorithms to enable efficient parallel simulations of a still wider class of large-scale stochastic models [35–38].

This work was supported by Defi NEEDS (Project MathDef) and Lawrence Livermore National Laboratory's LDRD office (Project No. 09-ERD-005) and utilized HPC resources from GENCI-[CCRT/CINES] (Grant No. x2013096973). This work was

performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract No. DE-AC52-07NA27344. The authors wish to express their gratitude to T. Opplestrup, F. Soisson, E. Clouet, J.-L. Bocquet, G. Adjanor, and A. Donev for fruitful discussions.

[1] N. G. van Kampen, *Stochastic Processes in Physics and Chemistry* (Elsevier Science, New York, 2007).
[2] S. Redner, *A Guide to First-Passage Processes* (Cambridge University Press, Cambridge, England, 2001).
[3] J.-M. Lanore, Radiat. Eff. **22**, 153 (1974).
[4] A. Bortz, M. Kalos, and J. Lebowitz, J. Comput. Phys. **17**, 10 (1975).
[5] D. Gillespie, J. Chem. Phys. **81**, 2340 (1977).
[6] P. Bolhuis, D. Chandler, C. Dellago, and P. Geissler, Annu. Rev. Phys. Chem. **53**, 291 (2002).
[7] S. X. Sun, Phys. Rev. Lett. **96**, 210602 (2006); **97**, 178902 (2006).
[8] B. Harland and X. Sun, J. Chem. Phys. **127**, 104103 (2007).
[9] C. D. Van Siclen, J. Phys. Condens. Matter **19**, 072201 (2007).
[10] N. Eidelson and B. Peters, J. Chem. Phys. **137**, 094106 (2012).
[11] T. Mora, A. M. Walczak, and F. Zamponi, Phys. Rev. E **85**, 036710 (2012).
[12] M. Manhart and A. V. Morozov, Phys. Rev. Lett. **111**, 088102 (2013).
[13] The evolution operator is obtained by integrating the ME $\dot{\mathbf{v}}^{\mathrm{T}}(t) = \mathbf{v}^T(t)\mathbf{M}$ from t to $t+\tau$ and identifying the formal solution with $\mathbf{v}^T(t+\tau) = \mathbf{v}^T(t)\mathbf{P}(t, t+\tau)$, where $\mathbf{v}^T(t)$ is the state-probability (row) vector at t.
[14] If the system is in β at a given time, then the state-probability vector at a later time τ is $\mathbf{v}_\beta^T(\tau) = \mathbf{1}_\beta^T\mathbf{P}(0,\tau)$, where $\mathbf{1}_\beta^T$ denotes the row vector whose βth entry is one and the other ones are zero. The entries of $\mathbf{v}_\beta^T(\tau)$ are $P_{\beta\gamma}(0,\tau)$ and can be used as transition probabilities for KMC moves from β.
[15] M. A. Novotny, Phys. Rev. Lett. **74**, 1 (1995).
[16] G. Boulougouris and D. Theodorou, J. Chem. Phys. **127**, 084903 (2007).
[17] M. Barrio, A. Leier, and T. Marquez-Lago, J. Chem. Phys. **138**, 104114 (2013).
[18] G. Nandipati, Y. Shim, and J. G. Amar, Phys. Rev. B **81**, 235415 (2010).
[19] C. Moler and C. Van Loan, SIAM Rev. **45**, 3 (2003).
[20] M. Athènes, P. Bellon, and G. Martin, Philos. Mag. A **76**, 565 (1997).
[21] S. Trygubenko and D. Wales, J. Chem. Phys. **124**, 234110 (2006).
[22] D. Wales, J. Chem. Phys. **130**, 204111 (2009).
[23] S. A. Serebrinsky, Phys. Rev. E **83**, 037701 (2011).
[24] See Supplemental Material at http://link.aps.org/supplemental/10.1103/PhysRevLett.113.230601 for the connection between the present path factorization and Gauss-Jordan elimination method, and for a detailed description of the algorithm.
[25] Y. Limoge and J.-L. Bocquet, Phys. Rev. Lett. **65**, 60 (1990).
[26] F. Soisson and C. C. Fu, Phys. Rev. B **76**, 214102 (2007).
[27] M. Athènes, P. Bellon, and G. Martin, Acta Mater. **48**, 2675 (2000).
[28] E. Clouet, in *Fundamentals of Modeling for Metals Processing*, ASM Handbook Vol. 22A, edited by D. U. Furrer and S. L. Semiatin (ASM International, Materials Park, Ohio, 2010), pp. 203—219.
[29] To understand the origin of the efficiency decrease, we have monitored the number of distinct shapes of the vacancy-copper cluster. For a V-Cu_{30} cluster, we found that, over the 10^3 last factorizations that have been performed, there are only 21 different cluster shapes and that the five most frequent shapes occur with a frequency of about 60%.
[30] L. K. Béland, P. Brommer, F. El-Mellouhi, J. F. Joly, and N. Mousseau, Phys. Rev. E **84**, 046704 (2011).
[31] S. Pogatscher, H. Antrekowitsch, M. Werinos, F. Moszner, S. S. A. Gerstl, M. F. Francis, W. A. Curtin, J. F. Löffler, and P. J. Uggowitzer, Phys. Rev. Lett. **112**, 225701 (2014).
[32] D. Mason, R. Rudd, and A. Sutton, Comput. Phys. Commun. **160**, 140 (2004).
[33] B. Puchala, M. Falk, and K. Garikipati, J. Chem. Phys. **132**, 134104 (2010).
[34] T. Opplestrup, V. V. Bulatov, G. H. Gilmer, M. H. Kalos, and B. Sadigh, Phys. Rev. Lett. **97**, 230602 (2006).
[35] Y. Shim and J. G. Amar, Phys. Rev. B **71**, 115436 (2005).
[36] M. Merrick and K. Fichthorn, Phys. Rev. E **75**, 011606 (2007).
[37] E. Martínez and P. R. Monasterio, and J. Marian, J. Comput. Phys. **230**, 1359 (2011).
[38] F. Wieland and D. Jefferson, in *Proceedings of the 1989 International Conference on Parallel Processing*, edited by F. Ris and M. Kogge, (Pennsylvania State Univ. Press, University Park, PA, 1989) Vol. III, pp. 255—258.