



**HAL**  
open science

**Absorption kinetics of vacancies by cavities in aluminum: Numerical characterization of sink strengths and first-passage statistics through Krylov subspace projection and eigenvalue deflation**

Savneet Kaur, Manuel Athenes, Jérôme Creuze

► **To cite this version:**

Savneet Kaur, Manuel Athenes, Jérôme Creuze. Absorption kinetics of vacancies by cavities in aluminum: Numerical characterization of sink strengths and first-passage statistics through Krylov subspace projection and eigenvalue deflation. *Journal of Computational Physics*, 2022, 454, pp.110987. 10.1016/j.jcp.2022.110987 . cea-03746277

**HAL Id: cea-03746277**

**<https://cea.hal.science/cea-03746277>**

Submitted on 5 Aug 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Absorption kinetics of vacancies by cavities in Aluminum: numerical characterization of sink strengths and first-passage statistics through Krylov subspace projection and eigenvalue deflation

Savneet Kaur<sup>a</sup>, Manuel Athènes<sup>a,\*</sup>, Jérôme Creuze<sup>b</sup>

<sup>a</sup>*Université Paris-Saclay, CEA, Service de Recherches de Métallurgie Physique, F-91191 Gif-sur-Yvette, France*

<sup>b</sup>*SP2M ICMMO, Université Paris-Saclay, 91405 Orsay, France*

---

## Abstract

Modeling the microstructural evolution of metal and alloys, specifically under irradiation, is essential to predict the aging properties of materials. Many models are based on a transition rate matrix describing the jump frequencies of defects and involve a master equation governing the time evolution of a state probability vector. Here, we present non-stochastic numerical techniques to characterize the motion of individual defects migrating over long distances prior to recombining or being absorbed by another defect, resorting to the theory of absorbing Markov chains. These important events are fully determined by their first-passage time distribution to distant locations, no-passage distribution, and walker fluxes to the sinks. We show that these functions can be efficiently computed using a method combining Krylov subspace projection and eigenvalue deflation. For a model system describing the absorption of a vacancy by a cavity in aluminum, the use of a small Krylov subspace deflated by the unique eigenmode corresponding to the quasi-stationary distribution is sufficient to capture the kinetics of the defect absorption faithfully. This method can be used in kinetic Monte Carlo simulations to perform stochastic non-local moves or in cluster dynamics simulations to compute sink strengths.

---

\*Corresponding author

*Email address:* `manuel.athenes@cea.fr` (Manuel Athènes)

*Keywords:* Sink strengths calculations; Absorbing Markov processes; First passage distributions; Krylov Subspace projection.

---

## 1. Introduction

Master equations are used extensively to describe many dynamical systems in natural and engineering sciences, in domains ranging from sociology, neuroscience, signal processing, and biochemistry to condensed matter physics. They involve a transition rate matrix and govern the evolution of a high dimensional state probability vector. Introductory examples for processes governed by a master equation are the random walk on an integer line with possible steps in forward, and backward directions, Brownian motion, the path of a diffusing particle until it gets absorbed [1, 2]. The time evolution of condensed matter systems can be simulated directly using one stochastic approach like Langevin dynamics for model systems whose phase space is continuous and Kinetic Monte Carlo (KMC) methods for discrete systems. Any simulation consists of randomly generating typical trajectories in the path space. Some of the physical systems that KMC involves are defect diffusion in alloys, for instance, vacancy motion in general or vacancy/interstitial clustering in ion or neutron-irradiated materials of nuclear reactors [3, 4]. While pressure vessels in industrial reactors are usually made of ferritic stainless steels, aluminum alloys are also employed in some experimental reactors. Under irradiation, these materials lose their mechanical properties over time due to the creation of vacancies and interstitials that recombine and form vacancy cavities and interstitial loops. Given its complexity, the master equation associated with the microstructural evolution [5, 6] of irradiated materials is often simulated using various KMC methods [7].

Several KMC methods are available and are implemented depending on the investigated time and space scales. Atomistic kinetic Monte Carlo (AKMC) methods monitor the positions of all atoms on the lattice [8–10] or over the space [11, 12]. The variant method is commonly known as object kinetic Monte Carlo (OKMC), rather simulate diffusing entities like defect-solute clusters [13].

The class of OKMC methods encompasses event Kinetic Monte Carlo (EKMC) methods [14, 15] and first-passage kinetic Monte Carlo (FPKMC) methods [16, 17]. The former assumes that all events are possible at any time independently of each other. Simultaneously, the latter introduces spatial protections and enforces the synchronization of the diffusing entities rigorously based on the exact first-passage and no-passage distributions defined hereafter.

Historically, FPKMC algorithms have been developed to speed up KMC simulations. Indeed, the traditional KMC method may become inefficient when employed to simulate all the hops of defects on a lattice [18] and when the transition rate matrix equation exhibits a broad spectrum of frequencies. The causes of inefficiency may be energetic or entropic in origin. In the former situation, a diffusing defect performs many transitions between a few atomic configurations connected by small energy barriers, typically a vacancy binding to a solute cluster, before escaping elsewhere [19]. These connected configurations form trapping basins. The typical escape time of a defect is much higher than the typical time to cross into the small barriers. Subsequently, the freed vacancy may perform a considerable number of hops in bulk before recombining with another defect or being absorbed: the entropic origin for simulation inefficiency refers to this situation. The statistically exact approach to mitigate the inefficiency of the KMC methods is to draw sequences of events and first-passage times based on the theory of absorbing Markov chains [16, 17, 20–24]. Mathematically, the first passage times are the sum of the residence times spent by the walker in connected states before getting absorbed by an artificial or physical sink. The absorbing sink is artificial when it corresponds to the peripheral states of an energetic trap, while a physical absorbing sink usually corresponds to an energetic trapping basin, like solute clusters and dislocations. Furthermore, once the system has reached an absorbing state in absorbing Markov chains, it stays there infinitely. Because the probability of being absorbed tends to one as time tends to infinity, the connected states of the trapping basin are commonly known as transient states. Besides, a defect initially located in any transient state can reach any absorbing state, but not necessarily in one step.

The no-passage distribution is the conditional probability distribution of the defect on the transient states knowing that it has not been absorbed yet: the sum of probabilities over the transient states is one.

Note that master equations associated with absorbing Markov chains also play a crucial role in the formulation and calculation of sink strengths [25–28]. Sink strengths are important input parameters controlling the reaction rates in cluster dynamics (CD) simulations [29]. CD equations consist of a high-dimensional set of ordinary differential equations describing the evolution of cluster concentrations. They are obtained through a coarse-graining procedure from the chemical master equation describing the probability vector of cluster populations. Sink strengths are inversely proportional to the mean first passage time and the diffusion coefficient [25, 30].

There exist several ways to characterize absorbing Markov chains numerically. The essential goal is to compute the first-passage and no-passage distributions. These two distributions serve to draw the first passage times and moves for a defect to reach the absorbing sink. To achieve these tasks, one implements one of the two following randomization procedures: kinetic path sampling or reverse sampling, based on the factorization [22, 31, 32] or the eigenvalue decomposition [20, 31] of the absorbing transition rate matrix, respectively. When eigenvalue decomposition is performed [20, 21, 23, 31], the survival probability of the defect on the transient state is computed directly, making it possible to draw the desired first-passage or no-passage times through reverse sampling. The original Markov chain describing the defect evolution is assumed to be reversible. This means that the transport of defects obeys detailed balance; the transition probability flux between any two states is invariant under Markov chain reversal. The transition rate matrix of the Markov chain can therefore be symmetrized using similarity transformation, which ensures the matrix to be symmetric negative semi-definite. The survival probability of the defect before getting absorbed is obtained from the transient evolution operator, a matrix exponential, which is the sum of decaying exponentials. The eigenspectrum of the matrix is real and strictly negative. In this work, the survival probability

distribution is estimated using the eigenvalue decomposition method.

In practice, it is unfortunately difficult to entirely factorize or diagonalize large sparse matrices using dense solvers based on Gaussian elimination, Givens rotations, or Householder reflections due to memory limitations. Krylov subspace projection (KSP) methods are commonly used to obtain solutions for sparse high-dimensional linear systems. The approximations to these solutions are estimated by minimizing the residual over the subspace formed. A well-known KSP method is the conjugate gradient (CG) [33], which is used for solving linear systems involving symmetric and positive definite matrices. For symmetric and possibly indefinite system, iterative method like minimum residual (MINRES) method is rather used [34]. In case of non-symmetric matrices, the biconjugate gradient stabilized (BiCGSTAB) method [35, 36] that is a generalized CG method, and also a generalized minimum residual (GMRES) method [36] are available. KSP methods can also be employed to extract a few pairs of eigenvectors and eigenvalues iteratively, for instance, the Krylov-Schur method [37, 38].

The eigenvector associated with the smallest eigenvalue is proportional to the quasi-stationary distribution (QSD). It corresponds to the eigenmode exhibiting the slowest decay and limiting the no-passage distribution in the asymptotic time limit [39, 40]. The QSD is observed to considerably contribute to the first-passage and no-passage distributions in applications [31]. It completely characterizes them when trapping is severe and has an energetic origin. However, for purely entropic traps, it is observed that many additional eigenmodes are necessary to capture the early stage absorption kinetics [31] correctly. Consequently, the computational cost increases with the number of significant eigenmodes. In this work, we investigate the ability of model order reduction techniques based on Krylov subspace projection and eigenvalue deflation to faithfully characterize the early-stage kinetics at a reduced cost, given an initial probability vector.

The article is organized as follows. In Section 2.1, we introduce the theory of absorbing Markov chains and then formalize the generalized eigenvalue problem. We next describe model order reduction techniques combining Krylov subspace

projection and eigenvalue deflation in Section 2.3. In Section 3, we assess the efficiency of the developed methods by applying them to two problems, a toy model for two-dimensional defect absorption and a realistic model describing the absorption of a distant single vacancy by a cavity in Aluminum [25]. We discuss the most efficient strategy to compute the probability vector at times shorter than the mean first-passage time depending on the problem. We conclude in Section 4.

## 2. Theory and methodology

### 2.1. Mathematical Formalism

A master equation (ME) is the set of ordinary differential equations describing the evolution of the state probability vector  $\mathbf{p}(t)$  and writes [24]:

$$\dot{\mathbf{p}}^T(t) = \mathbf{p}^T(t)\mathbf{K}, \quad (1)$$

where  $\mathbf{K}$  is the Markov matrix. Its elements  $K_{ij}$  are the transition rates from state  $i$  to state  $j$  and superscript  $T$  stands for the transposition. Here, transition rates are assumed to be time independent [22]. The time evolution operator obtained formally from solutions of the ME can be expressed as

$$\mathbf{P}(t, t') = \exp \left[ \int_t^{t'} \mathbf{K} dt \right] = \exp((t' - t)\mathbf{K}). \quad (2)$$

Matrix element  $P_{ij}(t, t')$  represents the probability to find a system in state  $j$  at time  $t'$  given that it was previously in state  $i$  at time  $t$ . For the models with large number of states, it is not possible to directly compute the solution from the matrix exponential. Nevertheless, this can be done for the absorbing Markov process coinciding with the original process on a subset of  $N$  states called transient states and herein indexed from 1 to  $N$ . States connected to the transient states are pooled together into a single state labeled as  $N+1$  and called absorbing state. The mathematical derivation below follows from Ref.[31]: the transition rate matrix associated with the absorbing Markov process is cast into

the following conventional form:

$$\mathbf{K}^a = \left( \begin{array}{c|c} -\mathbf{A}^a & \mathbf{A}^a \vec{\mathbf{1}} \\ \hline \vec{\mathbf{0}}^T & 0 \end{array} \right), \quad (3)$$

where  $\mathbf{A}^a$  is the  $N \times N$  matrix such that  $A_{ij}^a = -K_{ij}$ . We define  $\vec{\mathbf{1}} = (1, \dots, 1)^T$  as the  $N$ -dimensional column vector whose components all equal one and similarly  $\vec{\mathbf{0}} = (0, \dots, 0)^T$ . For the absorbing transition rate matrix  $\mathbf{K}^a$  as well, the sum of the elements of each row is equal to 0. The associated evolution operator is defined as,  $\mathbf{P}^a(t) = \exp[\mathbf{K}^a t]$ . Being a stochastic matrix, it satisfies the following relationship  $\forall i \in \llbracket 1, N+1 \rrbracket$ :

$$\sum_{j=1}^{N+1} P_{ij}^a(t) = 1. \quad (4)$$

Because the absorbing state  $N+1$  is taken into account, Eq.(4) entails the conservation of total probability. For any vector  $\boldsymbol{\nu}(t)$  evolving according to the ME

$$\dot{\boldsymbol{\nu}}^T(t) = \boldsymbol{\nu}^T(t) \mathbf{K}^a, \quad (5)$$

the probability to find the walker in one of the  $N+1$  states is also conserved:

$$\sum_{i=1}^{N+1} \nu_i(t) = 1. \quad (6)$$

State labeled  $N+1$ , the absorbing state, is the only recurrent state of the system, while all the other states are transient. Starting from initial state labeled  $i \leq N$ , the probability that the system is in transient state  $j \leq N$  is given by:

$$P_{ij}^a(t) = \mathbf{e}_i^T \exp(-\mathbf{A}^a t) \mathbf{e}_j, \quad (7)$$

where  $\mathbf{e}_i$  and  $\mathbf{e}_j$  are the standard basis vectors. We refer the reader to Ref. [31] for details concerning the derivation.

Here, we assume that the original Markov process is reversible, i.e. obeys the principle of detailed balance which implies that the forward and backward fluxes between any two states, labeled by  $i$  and  $j$ , are equal:

$$\rho_i K_{ij} = \rho_j K_{ji}, \quad (8)$$

where  $\rho_i$  and  $\rho_j$  denote the equilibrium stationary probabilities of the two involved states. When the condition given by Eq.(8) is satisfied, the state probability vector  $\boldsymbol{\rho}$  corresponds to an equilibrium Gibbs-Boltzmann distribution. For the two transient states  $i$  and  $j$ , the condition of detailed balance entails

$$\sqrt{\rho_i}A_{ij}^a/\sqrt{\rho_j} = \sqrt{\rho_j}A_{ji}^a/\sqrt{\rho_i}. \quad (9)$$

From Eq.(9), a symmetric matrix is defined:

$$A_{ij}^B = s_i s_j \sqrt{\rho_i} A_{ij}^a / \sqrt{\rho_i} = A_{ji}^B, \quad (10)$$

where  $s_i$  and  $1/\sqrt{\rho_i}$  are strictly positive scaling factors. For the sake of concision, we express the two scaling factors using diagonal definite positive matrices

$$\mathbf{S} = \sum_{i=1}^N s_i \mathbf{e}_i \mathbf{e}_i^T \quad \text{and} \quad \mathbf{R} = \sum_{i=1}^N \frac{1}{\sqrt{\rho_i}} \mathbf{e}_i \mathbf{e}_i^T. \quad (11)$$

Matrices  $\mathbf{S}$ ,  $\mathbf{R}$  and  $\mathbf{B} = \mathbf{S}^2$  commute and are invertible enabling one to define

$$\mathbf{A}^B = \mathbf{S}\mathbf{R}^{-1}\mathbf{A}^a\mathbf{R}\mathbf{S} = (\mathbf{S}\mathbf{R})^{-1}\mathbf{B}\mathbf{A}^a(\mathbf{R}\mathbf{S}). \quad (12)$$

Matrix  $\mathbf{B}$  is used as a preconditioner, i.e. it aims at decreasing the condition number of  $\mathbf{A}^B$ . Matrix  $\mathbf{S}\mathbf{R}$  serves to define a similarity transformation between  $\mathbf{A}^B$  and  $\mathbf{B}\mathbf{A}^a$  and to formulate a generalized symmetric eigenvalue problem. A similarity transformation preserving spectral properties, the eigenvalues of  $\mathbf{B}\mathbf{A}^a$  are real and definite positive. This is true in particular for  $\mathbf{A}^a$  when  $\mathbf{B}$  is set to the identity  $\mathbf{I}$ .

The spectrum of  $\mathbf{A}^a$  can be obtained by solving the following generalized eigenvalue problem:

$$\mathbf{A}^B \boldsymbol{\varphi}_k = \mathbf{B} \boldsymbol{\varphi}_k \lambda_k, \quad (13)$$

where  $\boldsymbol{\varphi}_k$  denotes the  $k$ th eigenvector associated with  $\lambda_k$ , the  $k$ th eigenvalue. We assume that eigenvalues are sorted in ascending order:  $0 < \lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_N$  and that eigenvectors are normalized. Hence,  $\boldsymbol{\Phi} = (\boldsymbol{\varphi}_1, \boldsymbol{\varphi}_2, \dots, \boldsymbol{\varphi}_N)$  is an orthonormal basis of eigenvectors. From Eq.(13), we obtain

$$\mathbf{A}^B \boldsymbol{\Phi} = \mathbf{B} \boldsymbol{\Phi} \boldsymbol{\Lambda}, \quad (14)$$

where  $\mathbf{\Lambda}$  is a diagonal matrix with eigenvalues as its diagonal elements ( $A_{ii} = \lambda_i$ ). Multiplying both the left and right sides by  $\mathbf{\Phi}^T$  results in,

$$\mathbf{\Phi}^T \mathbf{A}^B \mathbf{\Phi} = \mathbf{\Lambda}, \quad \mathbf{A}^B = \mathbf{B} \mathbf{\Phi} \mathbf{\Lambda} \mathbf{\Phi}^T \mathbf{B}, \quad (15)$$

where the  $\mathbf{B}$  orthogonality of  $\mathbf{\Phi}$  has been taken into account

$$\mathbf{\Phi}^T \mathbf{B} \mathbf{\Phi} = \mathbf{I}, \quad \mathbf{\Phi} \mathbf{\Phi}^T = \mathbf{B}^{-1}. \quad (16)$$

Now, the absorbing transition rate matrix can be obtained from Eq.(12) and Eq.(15) as

$$\mathbf{A}^a = \mathbf{R} \mathbf{S} \mathbf{\Phi} \mathbf{\Lambda} \mathbf{\Phi}^T \mathbf{S} \mathbf{R}^{-1}. \quad (17)$$

From the matrix decomposition (17), the exponential of the transition rate matrix is expressed as

$$\exp[-\mathbf{A}^a t] = \mathbf{R} \mathbf{S} \mathbf{\Phi} \exp[-\mathbf{\Lambda} t] \mathbf{\Phi}^T \mathbf{S} \mathbf{R}^{-1}. \quad (18)$$

We now introduce  $\mathbf{s}$  and  $\mathbf{r}$ , the scaling and re-scaling vectors consisting of diagonal elements of matrices  $\mathbf{S}$  and  $\mathbf{R}$ , respectively. We have  $\mathbf{r} = \mathbf{R} \tilde{\mathbf{1}}$  and  $\mathbf{s} = \mathbf{S} \tilde{\mathbf{1}}$  and also define two additional sets of re-scaled basis vectors  $\mathbf{g}_i = \mathbf{e}_i \odot \mathbf{s} \odot \mathbf{r}$  and  $\mathbf{d}_j = \mathbf{e}_j \odot \mathbf{s} \odot \mathbf{r}$ , where  $\odot$  and  $\oslash$  denote component-wise multiplication and division.

Plugging the scalar products  $g_i^h = \varphi_h^T \mathbf{g}_i$  and  $d_j^h = \varphi_h^T \mathbf{d}_j$  into the evolution operator, Eq.(7) gives

$$P_{ij}^a(t) = \sum_{h=1}^N g_i^h d_j^h \exp(-\lambda_h t), \quad (19)$$

where  $(i, j) \leq N$ . The survival probability of a system that has evolved from the initial state  $i$  at time  $t = 0$  is the probability that the system has not been absorbed yet, or equivalently, is still located in one of the transient state. From Eq.(19), it is easy to deduce

$$p_i^s(t) = \sum_{j=1}^N P_{ij}^a(t) = \sum_{j=1}^N \sum_{h=1}^N g_i^h d_j^h \exp(-\lambda_h t) = \sum_{h=1}^N \alpha_i^h \exp(-\lambda_h t), \quad (20)$$

where  $\alpha_i^h = g_i^h \sum_{j=1}^N d_j^h$ . In Section 3, the survival probability is simply denoted by  $S(t) = \boldsymbol{\pi}^T \mathbf{p}^s(t)$ , omitting the dependence on the initial state  $\boldsymbol{\pi}$ . The mean first-passage time (MFPT) vector,  $\boldsymbol{\tau}^{(N)} = \mathbf{R}\mathbf{S}\mathbf{A}^{-1}\mathbf{S}\mathbf{R}^{-1}\vec{\mathbf{1}}$ , corresponds to the solution of a linear system of  $N$  equations. Besides, the MFPT from initial state  $i$  exhibits a simple expression in the eigen basis:

$$\tau_i^{(N)} = \sum_{h=1}^N \alpha_i^h / \lambda_h. \quad (21)$$

## 2.2. Quasi-stationary distribution and residence times

The no-passage probability distribution is defined by  $P_{ij}^{\text{np}}(t) = P_{ij}^{\text{a}}(t)/p_i^{\text{s}}(t)$  for  $i, j \in \llbracket 1, N \rrbracket$ . Likewise, the QSD is defined as the probability vector  $q_j$  that behaves asymptotically in time as the no-passage distributions. This is independent of initial state  $i$

$$q_j = \lim_{t \rightarrow \infty} P_{ij}^{\text{np}}(t) = \frac{d_j^1}{\sum_{h=1}^N d_h^1} \quad (22)$$

The MFPT is obtained by integrating over time:

$$\tau_i^{(N)} = \int_0^\infty t \frac{d}{dt} p_i^{\text{a}}(t) dt = \mathbf{e}_i^T \mathbf{A}^{-1} \vec{\mathbf{1}}, \quad (23)$$

where the quantity  $p_i^{\text{a}}(t) = 1 - p_i^{\text{s}}(t)$  corresponds to the probability that the walker has been absorbed at  $t$ . The absorption probability  $p_i^{\text{a}}$  increases from 0 at  $t = 0$  to 1 at  $t = \infty$ . The time derivative of  $p_i^{\text{a}}(t)$  corresponds to the probability distribution of first-passage times.

## 2.3. Model Order Reduction and Iterative Methods

### 2.3.1. Eigenvector Subspace Model Projection

As mentioned in the introduction, it is not possible to compute the entire eigenspectrum of huge sparse matrices using standard dense solvers based on Givens rotation or Householder reflections for memory limitations. Instead, sparse and iterative eigenvalue solvers [41–43] are to be implemented to extract a limited portion of the eigenspectrum and to approximate the evolution operator by its projection on the reduced eigenbasis [31]. In practice, the basis

encompasses the  $k$  eigenvectors with smallest eigenvalues. By doing so, a reduction of the model is performed and the symmetric transition rate matrix  $\mathbf{A}^a$  in Eq.(17) is approximated by

$$\mathbf{A}^{(k)} = \mathbf{SR} \left[ \sum_{h=0}^k \varphi_h \varphi_h^T \lambda_h \right] \mathbf{SR}^{-1}, \quad (24)$$

and the evolution operator is approximated by

$$P_{ij}^{(k)}(t) = \sum_{h=1}^k g_i^h d_j^h \exp(-\lambda_h t). \quad (25)$$

In the following, we will refer to this approach as the Eigenvector Subspace Model Projection (ESMP) method. Assessing the convergence as a function of the eigenvalue number  $k$  can be done by calculating the survival probability on the reduced eigenvector space and comparing the result with the one obtained using a standard solver computing the matrix exponential function with a built-in convergence criteria [44]. An additional verification can be performed by comparing the truncated MFPT

$$\tau_i^{(k)} = \sum_{h=1}^k \alpha_i^h \lambda_h \quad (26)$$

to the exact one, i.e., to the value  $\tau_i^{(N)}$  defined in Eq. (21) and computed using two well-established and robust methods: conjugate gradient and multi-frontal Cholesky [45] algorithms.

In the following subsections, we present additional model reduction approaches based on Krylov subspace projection methods: refer to [35, Ch. 7] for a textbook. Notice that  $\mathbf{A} \in \mathbb{R}^{N \times N}$  will stand for  $\mathbf{A}^I$ , i.e. the scaling matrix  $\mathbf{B}$  defined in previous section for the sake of generality will always be equal to the identity to restrict our investigation.

### 2.3.2. Krylov Subspaces

Evaluating matrix functions via eigenvalue decomposition becomes extremely costly for large sparse matrices whenever many eigenpairs are needed. A solution to this problem is to restrict the computation to the product of a vector a

matrix function of  $\mathbf{A} \in \mathbb{R}^{N \times N}$  and a vector  $\mathbf{b} \in \mathbb{R}^N$ :

$$\mathbf{x} = f(\mathbf{A})\mathbf{b}. \quad (27)$$

This task is efficiently accomplished using a Krylov subspace projection (KSP) method [46]. In particular, KSP methods are among the most efficient algorithms for estimating the solution of huge sparse linear systems. As examples of Krylov subspace solvers, one mentioned in the introduction, the conjugate gradient and minimal residual methods aim at computing the quantity

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}, \quad (28)$$

which amounts to setting  $\mathbf{f}(\mathbf{A}) = \mathbf{A}^{-1}$  [35, Ch. 9]. The Krylov subspace is characterized by its dimension  $\ell$ , matrix  $\mathbf{A}$ , and vector  $\mathbf{b}$  and is denoted by  $\mathcal{K}_\ell(\mathbf{A}, \mathbf{b})$ . Its construction proceeds as follows: vector  $\mathbf{b}$  is left multiplied by  $\mathbf{A}$ , which results in a new vector  $\mathbf{A}\mathbf{b}$ . The new vector is multiplied again with matrix  $\mathbf{A}$  to find  $\mathbf{A}^2\mathbf{b}$ , and this goes on  $\ell - 1$  times, vector  $\mathbf{b}$  being included in the subspace:

$$\mathcal{K}_\ell(\mathbf{A}, \mathbf{b}) = \text{span}\{\mathbf{b}, \mathbf{A}\mathbf{b}, \mathbf{A}^2\mathbf{b}, \dots, \mathbf{A}^{\ell-1}\mathbf{b}\}. \quad (29)$$

Gram-Schmidt algorithm is also used on the fly to construct an orthogonal basis of  $\mathcal{K}_\ell(\mathbf{A}, \mathbf{b})$  whose dimension is always equal to  $\ell$ . If it happens that  $\mathbf{A}^l \in \mathcal{K}_l(\mathbf{A}, \mathbf{b})$ , the subspace construction is resumed and  $\ell$  is set to  $l$ . The basis of  $\mathcal{K}_\ell(\mathbf{A}, \mathbf{b})$  is denoted by  $\mathbf{V}_\ell$ .

### 2.3.3. Krylov Subspace Model Projection

Using the definition of Krylov subspace from Section 2.3.2, the standard scheme involving matrix-vector multiplications is recalled. In practice, the problem of computing the exponential of a huge sparse matrix  $\mathbf{A} \in \mathbb{R}^{N, N}$  is reduced to the one of computing the exponential of a small matrix  $\mathbf{T}_\ell$  of dimension  $\ell$

$$\mathbf{A}\mathbf{V}_\ell = \mathbf{V}_\ell\mathbf{T}_\ell + T_{\ell+1, \ell}\mathbf{v}_{\ell+1}\widehat{\mathbf{e}}_\ell^T, \quad (30)$$

where  $\mathbf{V}_\ell = [\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_\ell]$  consists of  $\ell$  orthonormal column vectors and  $\mathbf{T}_\ell \in \mathbb{R}^{\ell, \ell}$  is a symmetric tri-diagonal matrix. Since the vectors from  $\mathbf{V}_\ell$  basis

are orthonormal, we have  $\mathbf{V}_\ell^T \mathbf{V}_\ell = \mathbf{I}_\ell$  where  $\mathbf{I}_\ell \in \mathbb{R}^{\ell, \ell}$  is the identity matrix.  $\mathbf{T}_\ell$  corresponds to the projection of  $\mathbf{A}$  onto  $\mathbf{V}_\ell$ , and  $\widehat{\mathbf{e}}_\ell^T$  denotes the  $\ell$ th unit coordinate vector of  $\mathbb{R}^\ell$ . The reduced tridiagonal matrix is obtained by left-multiplying both sides of Eq.(30) by  $\mathbf{V}_\ell^T$ :

$$\mathbf{V}_\ell^T \mathbf{A} \mathbf{V}_\ell = \mathbf{V}_\ell^T \mathbf{V}_\ell \mathbf{T}_\ell = \mathbf{T}_\ell, \quad (31)$$

while the Arnoldi approximation of  $\mathbf{A}$  is  $\mathbf{V}_\ell \mathbf{T}_\ell \mathbf{V}_\ell^T$ . This last matrix is then used to approximate  $f(\mathbf{A})\mathbf{b}$  using the following vector

$$\mathbf{f}_\ell = f(\mathbf{V}_\ell \mathbf{T}_\ell \mathbf{V}_\ell^T) \mathbf{b} = \beta \mathbf{V}_\ell f(\mathbf{T}_\ell) \widehat{\mathbf{e}}_1, \quad (32)$$

where  $\widehat{\mathbf{e}}_1$  stands for the first unit coordinate vector of  $\mathbb{R}^\ell$ ,  $\beta$  for  $\|\mathbf{b}\|$ , the Euclidean norm of  $\mathbf{b}$ , and where we plugged the relation  $\mathbf{b} = \beta \mathbf{V}_\ell \widehat{\mathbf{e}}_1$ .

This approach is another example of model order reduction. For a sparse matrix, the complexity is  $\mathcal{O}(N\ell)$  for storage and  $\mathcal{O}(N^2\ell)$  for the  $\ell$  steps of iteration [47, 48]. Krylov subspaces  $\mathcal{K}_\ell(\mathbf{A}, \mathbf{b})$  of low dimensions are used in practice, hence this approach is considerably faster than exact full diagonalization techniques that exhibit  $\mathcal{O}(N^3)$  complexity in number of operations and  $\mathcal{O}(N^2)$  in storage.

#### 2.3.4. Eigenvector and Krylov Subspace Model Projection

In this section, we describe a third approach, referred to as eigenvector and Krylov subspace projection method (EKSMP), consisting in projecting the model both on eigenvector and Krylov subspaces. It is based on the standard deflation technique of linear algebra [49–51]. In deflation, the approximation subspace is divided into two complementary subspaces, so that the two parts of the solution are easier to find using an exact method in the first subspace and an iterative method in the second deflated subspace.

The deflation approach is usually implemented for solving linear systems but we consider it here for evaluating the application of any matrix function on a vector, i.e.  $f(\mathbf{A})\mathbf{b}$ . As previously, function  $f$  will be either the inverse or scaled exponential functions. The main objective of this scheme is to obtain a deflated

matrix  $\mathbf{A}^\perp$  whose condition number will be smaller than that of  $\mathbf{A}$ . This is done by discarding the contribution of a few smallest eigenvalues from the system, and focusing on a deflated matrix. Hence, the first subspace is generated by the  $k$  eigenvectors of  $\mathbf{A}$  associated with the lowest eigenvalues as in Section 2.3.1 and is denoted by  $\mathcal{E}_k(\mathbf{A})$ . The deflated subspace is the orthogonal component of the eigenvector subspace and is denoted by  $\mathcal{E}_k(\mathbf{A})^\perp$ . The Krylov subspace is then constructed in the deflated space. The goal is to accelerate the convergence of the projected dynamics towards the exact solution as the dimension  $\ell$  of the Krylov subspace increases.

An orthogonal basis of  $\mathcal{E}_k(\mathbf{A})$  writes  $\Phi_k = [\varphi_1, \varphi_2, \dots, \varphi_k]$ , entailing that  $\mathbf{P} = \Phi_k \Phi_k^T$  and  $\mathbf{I} - \mathbf{P}$  are the orthogonal projection operators on  $\mathcal{E}_k(\mathbf{A})$  and  $\mathcal{E}_k(\mathbf{A})^\perp$ . A general property of projection operators is that they are involution, i.e.  $\mathbf{P}^n = \mathbf{P}$ . A particular property of  $\mathbf{P}$  and  $\mathbf{I} - \mathbf{P}$  is that they commute with  $\mathbf{A}$ , as a result of the spectral theorem. Hence, any power of  $\mathbf{A}$  can be decomposed as

$$\mathbf{A}^n = (\mathbf{P}\mathbf{A})^n \mathbf{P} + ((\mathbf{I} - \mathbf{P})\mathbf{A})^n (\mathbf{I} - \mathbf{P}) = (\mathbf{A}^\parallel)^n \mathbf{P} + (\mathbf{A}^\perp)^n (\mathbf{I} - \mathbf{P}), \quad (33)$$

where  $\mathbf{A}^\parallel = \mathbf{P}\mathbf{A}$ ,  $\mathbf{A}^\perp = \mathbf{A} - \mathbf{A}^\parallel$ . Consequently, the desired quantity can be decomposed as the sum of the following two terms:

$$f(\mathbf{A})\mathbf{b} = f(\mathbf{A}^\parallel)\mathbf{b}^\parallel + f(\mathbf{A}^\perp)\mathbf{b}^\perp, \quad (34)$$

where  $\mathbf{b}^\parallel$  and  $\mathbf{b}^\perp$  stands for  $\mathbf{P}\mathbf{b}$  and  $(\mathbf{I} - \mathbf{P})\mathbf{b}$ , respectively. Note that the Krylov subspace in the deflated space can be simply generated from the projected initial vector  $(\mathbf{I} - \mathbf{P})\mathbf{b}$ , which is formally stated by

$$\mathcal{K}_\ell(\mathbf{A}^\perp, \mathbf{b}^\perp) = \mathcal{K}_\ell(\mathbf{A}, \mathbf{b} - \mathbf{P}\mathbf{b}). \quad (35)$$

As a result, the approximation subspace is the sum of two subspaces:

$$\mathcal{G}_{k,\ell}(\mathbf{A}, \mathbf{b}) = \mathcal{E}_k(\mathbf{A}) \oplus \mathcal{K}_\ell(\mathbf{A}, \mathbf{b} - \mathbf{P}\mathbf{b}), \quad (36)$$

which are orthogonal to each other. Note that ESMP method considers the first subspace only [Section 2.3.1], while KSMP method the second one only [Section 2.3.3]. The second term in Eq.(34) is projected onto the orthogonal Krylov

subspace  $\mathcal{K}_\ell(\mathbf{A}^\perp, \mathbf{b}^\perp)$  and evaluated using full eigenvalue decomposition. The matrix that must be diagonalized is

$$\mathbf{T}_\ell^\perp = (\mathbf{V}_\ell^\perp)^T \mathbf{A} \mathbf{V}_\ell^\perp, \quad (37)$$

where  $\mathbf{V}_\ell^\perp$  is the standard orthogonal basis of  $\mathcal{K}_\ell(\mathbf{A}^\perp, \mathbf{b}^\perp)$ . The accuracy of the model order reduction method can be verified by checking the convergence of survival probability distribution and estimation of MFPTs as a function of  $\ell$  given  $k$ . The survival probability of a system that has evolved from initial probability vector  $\boldsymbol{\pi}$  at time  $t = 0$  evaluated using EKSMP method is

$$S_{k,\ell}(t) = \sum_{i=1}^N \sum_{h=1}^k \pi_i \alpha_i^h \exp(-\lambda_h t) + \sum_{i=1}^N \sum_{h=1}^\ell \pi_i \hat{\alpha}_i^h \exp(-\lambda_h^\perp t) \quad (38)$$

where  $\lambda_h^\perp$  is the  $h$ th eigenvalue of  $\mathbf{T}_\ell^\perp$  and the weighting coefficient  $\hat{\alpha}_i^h$  involves the corresponding  $h$ th eigenvector  $\hat{\boldsymbol{\varphi}}_h$  in the Krylov subspace. After projecting with operator  $\hat{\boldsymbol{\varphi}}_h^T \mathbf{V}_\ell^T$ , we obtain  $\hat{\alpha}_i^h = \hat{g}_i^h \hat{d}^h$  with  $\hat{g}_i^h = \hat{\boldsymbol{\varphi}}_h^T \mathbf{V}_\ell^T \mathbf{s} \odot \mathbf{r} \odot \mathbf{e}_i$  and  $\hat{d}^h = \hat{\boldsymbol{\varphi}}_h^T \mathbf{V}_k^T \mathbf{s} \odot \mathbf{r}$ . Note that assuming exact arithmetic, the survival probability  $S(t) = \boldsymbol{\pi}^T \mathbf{p}^s(t)$  defined from Eq.(20) is equal to  $S_{N-\ell,\ell}(t)$  in Eq.(38),  $\forall \ell \in \llbracket 0, N \rrbracket$ , since the approximation space  $\mathcal{G}_{N-\ell,\ell}(\mathbf{A}, \boldsymbol{\pi})$  spans the entire phase space.

To later monitor the convergence of the methods as a function of  $k$  and  $\ell$ , we will first inspect the estimated survival probability at  $t = 0$  and additionally evaluate the following reduced MFPT:

$$T_{k,\ell} = \frac{\sum_{i=1}^N \pi_i \left[ \sum_{h=1}^k \alpha_i^h / \lambda_h + \sum_{h=1}^\ell \hat{\alpha}_i^h / \lambda_h^\perp \right]}{\sum_{i=1}^N \pi_i \tau_i^N}, \quad (39)$$

where the denominator in Eq.(39) corresponds to the MFPT from initial distribution  $\boldsymbol{\pi}$ . The MFPT's have been computed using two distinct linear solvers (sparse Cholesky and CG) to check that matching results are obtained.

#### 2.4. Implementation

The methods described above have been coded in PETSC/SLEPc environments resorting to the matrix function (MFN) object [41–43, 52–54]. MFN object also provides the restarted Krylov subspace projection method (R-KSP) [44]

to compute the product of common matrix functions and a vector. R-KSP was used to compute reference values to check the correctness of the faster methods described above. R-KSP is a robust and well-established KSP solver with built-in convergence criteria [44, 53] wherein the Krylov basis is restarted until a convergence criterion is fulfilled. We refer the reader to Appendix A for details. In our case, the application of vector  $\mathbf{b}$  on the matrix exponential is computed for a predetermined set of times  $\{t_n\}_{0 \leq n \leq L}$ :

$$\mathbf{x} = \exp(-\mathbf{A}t_n)\mathbf{b}. \quad (40)$$

The operation must be repeated at each considered time. For this purpose, R-KSP method is much more expensive than EKSMP method, because it is not able to provide the entire first-passage law at once.

### 3. Results and Discussions

In this section, we first discuss the scalability and efficiency of the four different solvers used to extract eigenvalues. We set up a simple absorption model in two dimensions in Section 3.1. Next, we illustrate the three computational methods discussed in Section 2.3 by applying them to a realistic problem, the absorption of a single vacancy in a cavity in aluminum in Section 3.2. The model describing thermally activated jumps of aluminum atoms into a next nearest-neighbor vacancy is detailed in Ref. [31]. It accounts in particular for the dipole-dipole elastic interactions between the vacancy and the cavity [25]. We characterize the absorption kinetics of the vacancy around the cavity and quantify the effect of the elastic interactions on the vacancy flux towards the cavity in Sec. 3.3.1, and on sink strengths (Sec. 3.3.2).

#### 3.1. Efficiency and scalability of eigensolvers

The simple absorption model describes the motion of a defect on a periodically replicated square lattice of size  $L$  and coordination number  $Z = 4$ . The defect hops from any site to any of its four nearest neighboring sites with reduced frequency of 1. The number of transient states is  $N = L^2$ . In addition,

the defect can reach the absorbing sink from a singularized site with an absorbing frequency equal to  $10^{-2}$ . In this particular application, the considered transition rate matrix  $\mathbf{A}$  is thus a modified Laplacian matrix: diagonal elements are equal to 4, but one element is set to value 4.01, while the  $4N$  off-diagonal elements corresponding to transitions are all equal to -1. The matrix  $\mathbf{A}$  is thus symmetric positive definite by construction. We herein evaluate the cost of extracting the linear system using the various sparse iterative solvers from SLEPC library [41] in PETSC environment [42] and with varying the number of cores in the computations. This enables one to deduce the speedup and efficiency resulting from implementing the solvers on a parallel computer architecture.

We use an in-house cluster composed of 36 processing nodes, each node consisting of a dual-processor Intel(R) Xeon(R) Gold 6132 CPU running at 2.60GHz with  $2 \times 14$  cores. As for the sparse iterative solvers, we tested methods like Krylov-Schur (KS), locally optimal block preconditioned conjugate gradient (LOBPCG), generalized Davidson (GD), and Jacobi Davidson (JD). In this first series of computations, the lattice size  $L$  is set to the value of  $10^2$ , so the number of transient states is  $N = 10^4$ . The eigenvalues to be considered sufficiently accurate and converged, the tolerance value to  $10^{-10}$ . The maximum number of iterations was set to  $6 \times 10^4$ . KS and LOBPCG solvers are implemented without preconditioning, while block Jacobi preconditioning is used by GD solver and also JD solver within its built-in MINRES routine. Figure 1 displays the CPU time required to compute the indicated number of eigenvalues using the four iterative solvers. We observe that the Krylov-Schur solver takes less time to extract the eigenvalues than LOBPCG, JD, and GD solvers do.

The performance of KS, LOBPCG, JD, and GD solvers are all expected to be positively impacted through preconditioning. We study the impact of preconditioning resorting to the very efficient sparse Cholesky solver from MUMPS, and report results for KS solver only in the following. Unfortunately, MUMPS preconditioner could not be enabled within LOBPCG in SLEPC, or sub-performed with JD and GD solvers.

For extracting the lowest eigenvalues using the preconditioned KS solver,

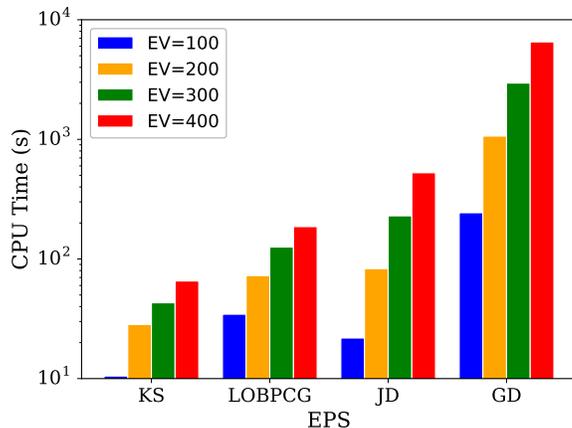


Figure 1: Comparison of CPU cost for four different iterative solvers (EPS) based on extracting the indicated number of requested eigenvalues (EV).

the inverse spectral transform is enabled and the linear system arising at each reversed iteration is solved using the Cholesky method of MUMPS. Cholesky factorization [55, 56] is done once using the MUMPS package [45]. We check the performance of Krylov-Schur with sparse Cholesky preconditioning by extracting 100 eigenvalues with varying the lattice size  $L$ . The tolerance parameter to extract converged eigenvalues is set to be  $10^{-10}$ . A linear speedup can be seen in Fig. 2 for the performance of Krylov-Schur if implemented with Cholesky preconditioning.

As the number of transient states  $N$  increases, it gets computationally more expensive to perform a sequential simulation to extract the eigenvalues. To check the scalability of the Krylov-Schur method, we performed parallel computations with 28 cores to extract the first smallest 100 eigenvectors and corresponding eigenvalues for the transient states  $N$  increasing from  $10^2$  to  $10^6$ . In Fig. 3, we observe a linear decrement in the cost of extracting eigenvalues as the number of cores and the number of transient states  $N$  are increased. It can be observed the complexity of the Krylov-Schur method is between square and linear.

In addition to this, we also calculated the speedup ratio and efficiency, de-

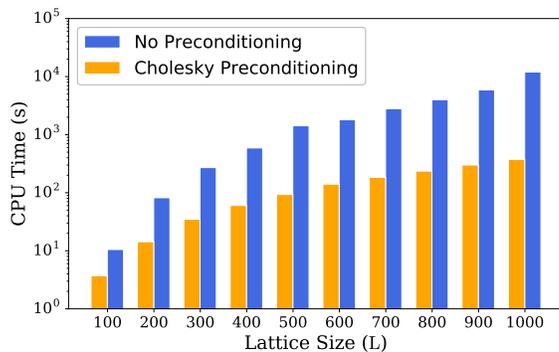


Figure 2: CPU times for extracting 100 eigenvalues using Krylov-Schur method without preconditioning and with Cholesky preconditioning as a function of the cell size ( $L$ )

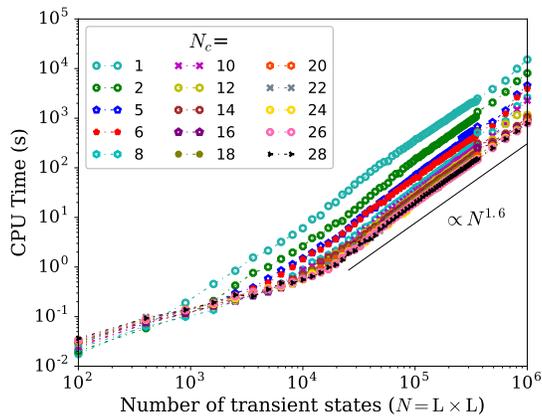


Figure 3: Evaluated cost for extracting 100 eigenvalues using Krylov-Schur method in sequential and parallel computations as a function of number of transient states  $N = L \times L$  with varying the number of cores ( $N_c$ ).

finned respectively by

$$S(N_c) = \frac{T(1)}{T(N_c)}, \quad \eta = \frac{S(N_c)}{N_c}, \quad (41)$$

where  $N_c$  corresponds to the number of cores. Again, the tolerance parameter was set to  $10^{-10}$  to provide a reasonable trade-off between accuracy and performance. We display in Fig. 4 the speedup ratio and efficiency for the following

set of transient states numbers:  $N = 10^2, 9 \times 10^4, 36 \times 10^4$  and  $10^6$ . The scalability efficiency  $\eta(N_c)$  decreases as the number of cores increases. However, as the number of transient states increases, the efficiency increases. As shown in Fig. 4, we observe that there is a deviation from the ideal linear speedup for  $N_c$  greater than 15. This results from the loss of communication between the processors. In the following, we found a similar scalability for the more realistic model discussed next and that a sequential implementation of the preconditioned KS solver was sufficient to solve our eigenvalue problems.

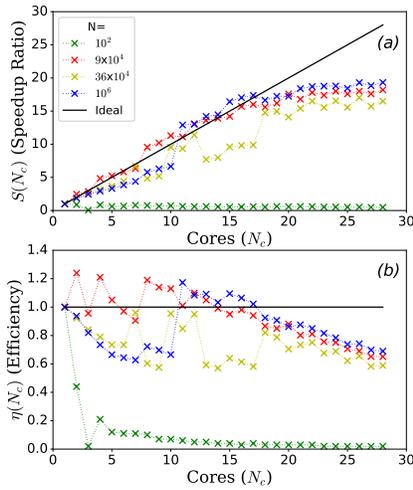


Figure 4: Computed (a) Speedup ratio and (b) Efficiency of Krylov-Schur method for varying transient states ( $N = L \times L$ ) as a function of cores ( $N_c$ ).

### 3.2. Survival Probabilities

We further assess the efficiency of the various subspace projection algorithms in computations involving the absorption of a vacancy by a cavity in aluminum. As in the previous two-dimensional model system, the tri-dimensional lattice cell is periodically replicated, entailing that an array of cavities is indeed involved. The cavity concentration is determined by the cell lengths of the lattice. This second model is more realistic as it accounts for the deformation field created by

the spherical cavity whose radius is  $20.7\text{\AA}$ . Temperature is 60 K. The evolution operator given by Eq.(19) provides us with the survival probability distribution Eq.(20) at time  $t$ . The goal is to characterize the absorption kinetics of a single vacancy around a cavity at 600K. The coordination number ( $Z = 12$ ) for the FCC lattice indicates the degree of matrix sparsity, i.e., a maximum of  $Z + 1$  elements per row are non zero. This model consists of 34801 transient states. We consider that the initial probability distribution of the vacancy is either localized or uniform: in the first setup, the vacancy is located at a distance of  $57.98\text{\AA}$  from the cavity center along the  $[110]$  direction. The second setup reflects the homogeneous creation of vacancies under irradiation, neglecting spatial correlations originating from cascades. In KMC simulations, the defect position would be drawn from the uniform multinomial law.

The survival probability distribution are computed using both setups of initial distributions, resorting to Eq.(20). Again, the tolerance parameter was set to  $10^{-12}$  and computations are performed using SLEPC and PETSC softwares [41, 52–54].

We display in Fig. 5(a) the survival probability distribution computed using the R-KSP method [Appendix A] as well as the ESMP method [Section 2.3.1] for up to the 3500 eigenvalues. For the ESMP method, we extracted the lowest eigenvalues by implementing the Krylov-Schur method with the tolerance parameter set to  $10^{-12}$ . For the R-KSP method, the tolerance parameter set to  $10^{-10}$  for better convergence. We extracted these lowest eigenvalues by implementing the Krylov-Schur method. The displayed distributions are scaled using the MFPT, estimated independently using the two standard linear solvers (sparse Cholesky and conjugate gradient). As it is observed from Fig. 5(a), the initial survival probability obtained from the ESMP method is not equal to one. This discrepancy is attributed to the fact that a substantial number of eigenmodes, higher than 3500, significantly contributes to the short time kinetics. The cost of extracting a huge portion of the eigen spectrum limits the applicability of the ESMP method. As for the R-KSP method, the survival probability is equal to one, but one needs to specify the Krylov subspace dimension. Typi-

cally, the value for the dimension of the Krylov subspace should not be less than 100. The number of restarted iterations to converge depends on the subspace dimension.

We display in Fig. 5(b) CPU times versus physical times for the R-KSP method. We observe that the simulation requires only 0.9 seconds for short time kinetics. However, this method becomes less efficient as time increases. It requires four more orders of magnitude of CPU time to converge at times larger than MFPT. It entails that R-KSP can be practically implemented for the evaluation of survival probabilities at short times only. We now investigate the range of applicability of KSMP and EKSMP methods and check whether they exhibit the same limitations as the ESMP method. The R-KSP method provides the reference data to support the EKSMP and KSMP algorithms.

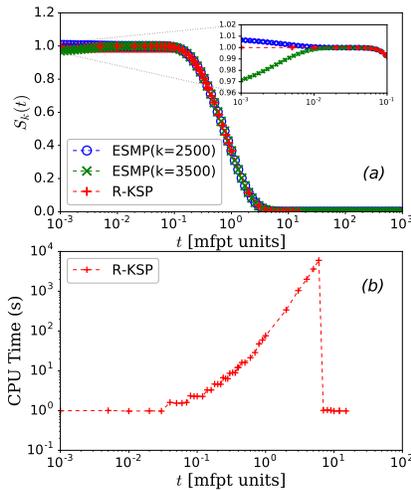


Figure 5: (a) Survival probability distribution computed using ESMP (2500 and 3500 eigenvalues), and R-KSP plotted in blue, green, and red, respectively. (b) CPU cost for computing survival probability vector at a particular instant of time using R-KSP.

The survival probability distributions estimated using KSMP and EKSMP methods for localized initial distribution are displayed in Fig. 6. The full eigenvalue decomposition of the reduced matrix  $\mathbf{T}_\ell$  was performed using a dense

solver from LAPACK library. The Krylov subspace dimension  $\ell$  is varying. We observe a fast convergence with respect to the Krylov subspace dimension for EKSMP and KSMP methods. Interestingly, the EKSMP method with  $k = 1$  requires a substantially smaller Krylov subspace, about 50 to 100, than the KSMP method does, about 500. This implies that the extra dimension required by the KSMP method aims at capturing the long-term kinetics of the QSD mode. This trend is more pronounced when the initial distribution is uniform, as observed in Fig. 7 wherein the survival probability and first passage distribution are displayed using KSMP and EKSMP methods and the same setups. This feature is attributed to the higher overlap between the initial distribution and the QSD. The dependence on initial conditions is the main limitation of EKSMP and KSMP methods, compared with the ESMP method that is less sensitive to the initial probability distribution. The significant number of non-local eigenmodes that must be computed contribute to the short time kinetics for a whole range of initial conditions. ESMP is, however, much more costly in terms of memory and CPU, as shown hereafter.

The CPU time taken by the three methods to calculate the survival probability distributions for the absorption problem are reported in Table 1. We used two different processors, referred to as processors A and B and whose features are given in the Table's caption. Sequential runs were performed with processors A and B. Parallel runs of ESMP method were also performed with processor A with 28 cores, yielding a speedup by a factor of 15 similar the one reported in Fig. 4 for the parallel simulations of the two-dimensional model. The ESMP method is computationally more expensive than EKSMP and KSMP methods. Furthermore, it is observed that EKSMP takes more CPU time than KSMP method does. It is because EKSMP method must extract the slowest eigenmode corresponding to the QSD, which requires 153 seconds of computational time. However, if one is interested in the absorption kinetics from a set of initial vacancy positions, then the EKSMP method will be more efficient because the QSD is computed once.

We quantified the reduced MFPT  $T_{k,\ell}$  and survival probability  $S_{k,\ell}(0)$  esti-

Processor	Methods	Condition	CPU Time (s)	Cores ( $N_c$ )
Proc-A	ESMP	$k = 3500, \ell=0$	$3.02 \cdot 10^4$	1
		$k = 3500, \ell=0$	$2.13 \cdot 10^3$	28
Proc-B	ESMP	$k = 1, \ell=0$	$8.93 \cdot 10^1$	1
		$k = 3500, \ell=0$	$2.96 \cdot 10^4$	1
	KSMP	$k = 0, \ell=50$	$1.10 \cdot 10^1$	1
		$k = 0, \ell=500$	$8.60 \cdot 10^1$	1
	EKSMP	$k = 1, \ell=50$	$9.83 \cdot 10^1$	1
		$k = 1, \ell=500$	$2.35 \cdot 10^2$	1
ESMP-CP	$k = 3500, \ell=0$	$4.66 \cdot 10^3$	1	
EKSMP-CP	$k = 1, \ell=500$	$1.70 \cdot 10^2$	1	

Table 1: CPU time (CPU) taken by ESMP, KSMP, and EKSMP methods to compute the first passage distributions for the single vacancy absorption. Processor A is Intel(R) Xeon(R) Gold 6132 CPU running, each node running at 2.60GHz with  $2 \times 14$  cores. Processor B is Intel i5-8400H running at 2.5GhZ with eight cores. CP stands for Cholesky preconditioning using MUMPS package and  $N_c$  is the used number of cores.

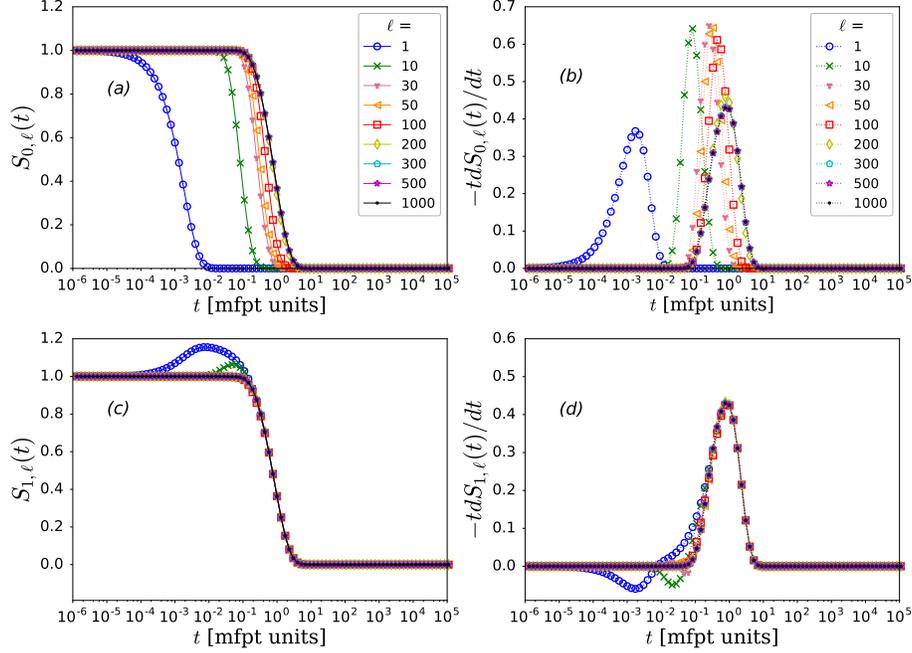


Figure 6: Comparison of survival probabilities and first passage distributions evaluated for the localized initial distribution using the KSMP method is presented in panels (a,b), and the EKSM method is presented in panels (c,d).

ated using the different methods to monitor the convergence as a function of  $k$  and  $\ell$ . The results are displayed in Fig. 8. The blue curve corresponds to the ESMP method. The 3500 eigenvalues evaluated previously have been used. The green and red curves correspond to EKSM and KSMP for  $l = 500$ , respectively. The reduced MFPT converges within  $2 \cdot 10^{-3}$  after  $\ell = 200$  for EKSM. Truncation errors for estimating the MFPTs are lower with a monotonous behavior and a fast convergence for the KSMP and EKSM methods as compared to the ESMP method. In panel (c) of Fig. 8, we clearly distinguish the convergence rates of the three methods.

To sample the MFPT starting from a uniform initial distribution, we used the EKSM method. The quantity  $T_{k,\ell}$  was computed with  $k = 1$ , and the results are displayed in Fig. 9. Convergence is clearly much faster when the initial distribution is uniform than when it is localized. Hence, a smaller Krylov

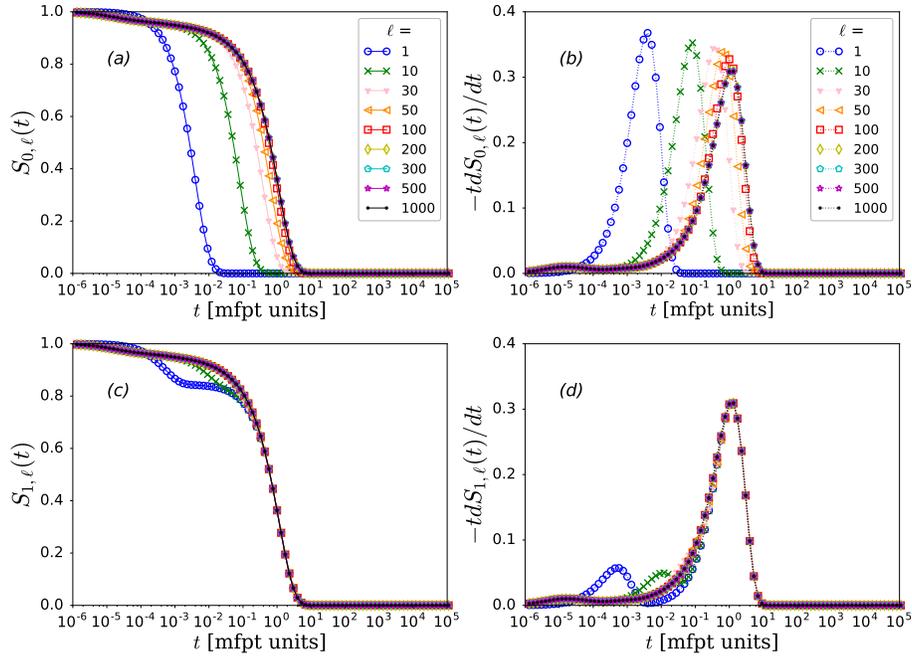


Figure 7: Comparison of survival probabilities and first passage distributions evaluated for the uniform initial distribution using the KSMP method is presented in panels (a,b), and the EKSM method is presented in panels (c,d).

subspace is required to evaluate the first passage distributions with accuracy and at a low computational cost. The main argument for using both Krylov subspace and eigenvector subspace projections is to reduce the dimension of the former subspace and to reuse the second subspace in other calculations. In Appendix B, additional results are reported concerning the use of Cholesky preconditioning (CP) and about the relevance of increasing the eigenvalue subspace  $k$  in EKSM method. It is shown that CP decreases the overall CPU cost and facilitates the extraction of additional eigenpairs. It should therefore be enabled whenever possible.

### 3.3. Vacancy absorption kinetics

To visualize the absorption kinetics, we compute and display the probability fluxes to the cavity and the sink strengths from the initial sites of the mobile

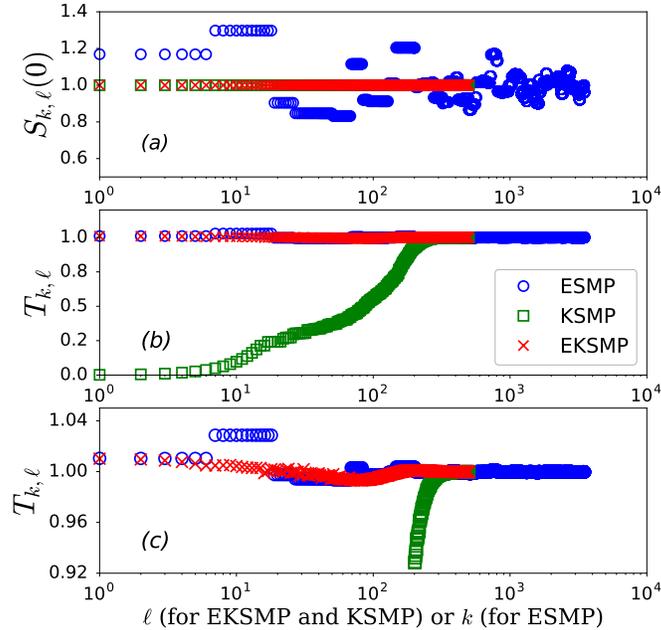


Figure 8: Comparison of (a) survival probabilities  $S_k(0)$ , (b) reduced MFPT  $T_{k,\ell}$  computed for ESMP (3500 eigenvalues), KSMP (500 eigenvalues) and EKSMP (500 eigenvalues) methods. Panel (c) zooms over values of the reduced MFPT ranging from 0.92 to 1.04 for the respective methods.

vacancy. The goal is to investigate the effect of the elastic deformation on the vacancy pathway to cavity.

### 3.3.1. Vacancy flux to cavity

We first computed the mean residence time vector  $\boldsymbol{\theta}$  defined by

$$\boldsymbol{\theta}^T = \boldsymbol{\pi}^T (\mathbf{A}^a)^{-1}, \quad (42)$$

for initial distribution  $\boldsymbol{\pi}$  by casting this equation in the form of Eq. (28) and using both the sparse Cholesky and CG solvers (to check that results were matching). Then, introducing the three dimensional lattice coordinates  $\hat{\mathbf{r}}_j$  of

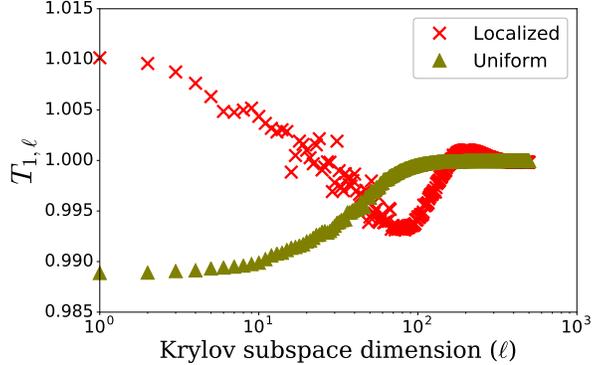


Figure 9: Reduced and truncated MFPT  $T_{1,\ell}$  evaluated for a localized and uniform initial distribution using EKSMP method.

the vacancy for state  $j$  [31], the vacancy flux was computed from the relation

$$\hat{\phi}_j = \frac{1}{2v} \sum_{\ell} (\theta_j K_{j\ell} - \theta_{\ell} K_{\ell j}) (\hat{\mathbf{r}}_{\ell} - \hat{\mathbf{r}}_j) \quad (43)$$

where  $\theta_j K_{j\ell} - \theta_{\ell} K_{\ell j}$  are the mean probability currents between both the transient and absorbing states,  $\boldsymbol{\theta}$  is the mean residence time vector and  $v$  represents the unit cell volume assumed to be uniform over the simulation box. All panels in Fig. 10 represents a quarter of the (001) plane containing the center of the cavity. The vacancy resides on the (100,100,0) Cartesian coordinates in Fig. 10, along  $\langle 110 \rangle$  direction. We computed residence times Eq.(42) and vacancy fluxes Eq.(43) for the localized initial distribution for sites  $j$  using linear solver. The algorithm used to compute these quantities are detailed in [31]. Figure 10.(a) represents the scaled residence times. We observe that the residence time is high at the periphery and low near the center, where the vacancy is more easily absorbed. Besides, anisotropy in the residence times can be observed as the vacancy evolves through the system and stays for shorter times along [100] and [010] directions, as observed in Fig. 10(b). The anisotropy in radial fluxes can also be observed in Fig. 10(c). Absorption path along the crystalline direction [110] depicts the anisotropic behavior.

The computed vacancy fluxes and residence times for the uniform distribution are displayed in Fig. 11. The trends are qualitatively similar to those

observed when the initial distribution was localized, but not quantitatively. In this setup, the residence times for vacancy at each site are less, implying that vacancy absorption happens faster.

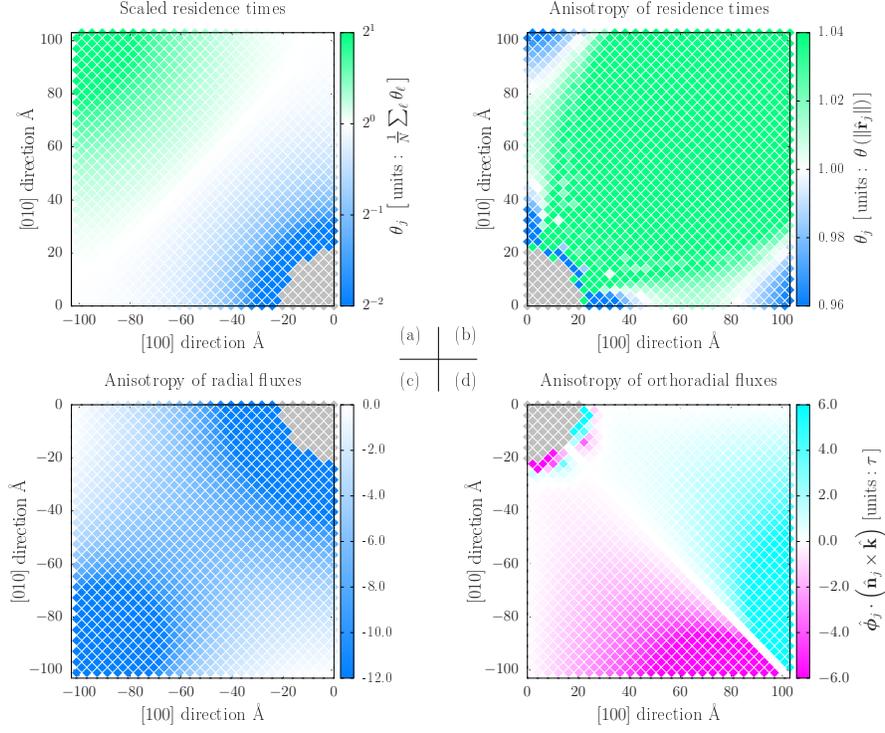


Figure 10: Estimation of residence times (a,b), radial vacancy flux (c), and ortho-radial vacancy fluxes (d) starting from a localized initial distribution. Absorption of a single vacancy initially located at a distance of 57.98 Å from the cavity center in  $\langle 100 \rangle$  crystalline directions. Coordinates of displayed sites satisfy  $\hat{\mathbf{r}}_j \cdot \hat{\mathbf{k}} = 0$  where  $\hat{\mathbf{k}}$  is the normalized basis vector orthogonal to (001).

Replacing the reduced residence time vector  $\boldsymbol{\theta}/\boldsymbol{\theta}^T \vec{\mathbf{1}}$  with the quasi-stationary probability vector  $\mathbf{q}$  in Eq.(43), provides the fluxes in the asymptotic time limit. The results are displayed in Fig. 12. The radial dependence of the QSD is shown in Fig. 12(a). The reduced probability decreases from 2 far from the cavity to 0.25 at the cavity periphery where the vacancy is about to get absorbed. The anisotropic nature of the radial and ortho-radial fluxes can also be observed in

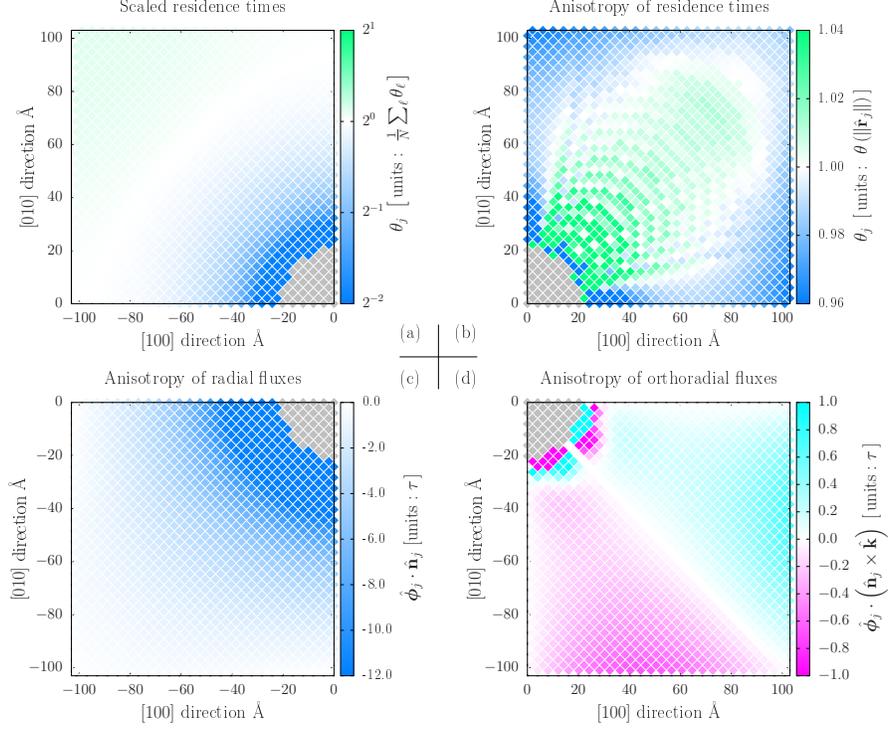


Figure 11: Estimation of residence times (a,b), radial vacancy flux (c), and ortho-radial vacancy fluxes (d) using a uniform initial distribution. The cavity center is in  $\langle 100 \rangle$  crystalline directions. Coordinates of displayed sites satisfy  $\hat{\mathbf{r}}_j \cdot \hat{\mathbf{k}} = 0$  where  $\hat{\mathbf{k}}$  is the normalized basis vector orthogonal to (001).

Fig. 12 (c,d). The more pronounced anisotropic behavior observed in Fig. 10 is due to the localized initial distribution, whereas anisotropy is less critical in Fig. 11 and Fig. 12. The residual anisotropy associated with the QSD is entirely due to the presence of the elastic field created by the cavity.

### 3.3.2. Sink strengths

We next quantify the sink strengths defined by

$$k^2 = \frac{1}{\tau D_v} = \frac{N}{\sum_{j=1}^N \tau_j^{(N)} D_v} \quad (44)$$

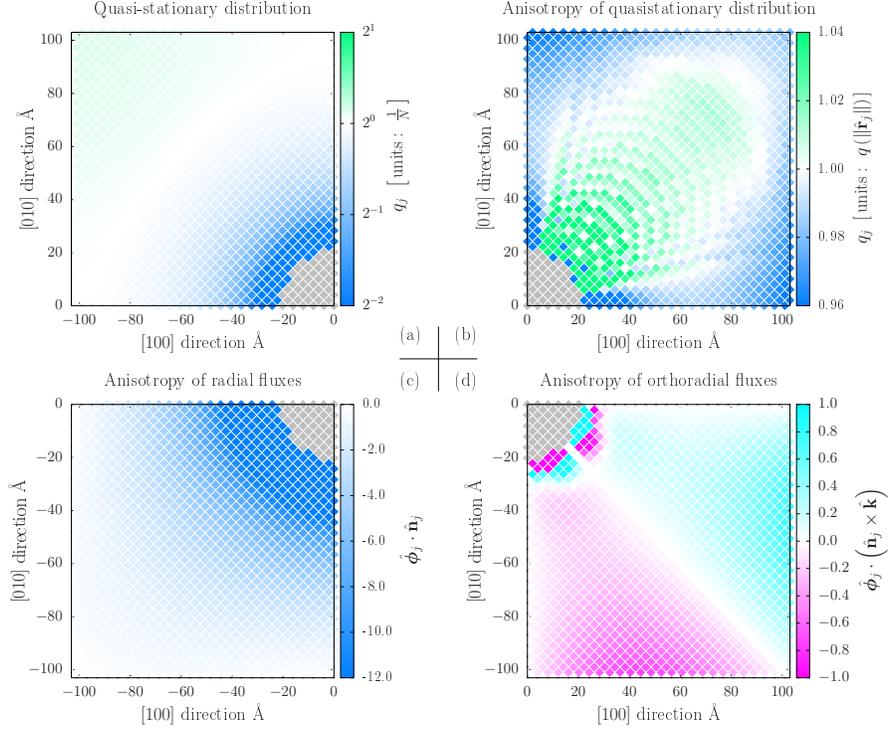


Figure 12: Estimated quasi-stationary distribution (a), anisotropy of quasi-stationary distribution probability (b), anisotropy of radial vacancy fluxes (c), and anisotropic ortho-radial vacancy fluxes (d) for the localized initial distribution.

where  $\tau$  denotes the MFPT associated with the uniform distribution and  $D_v$  the diffusion coefficient of the vacancy. Letting  $\nu$  denote the vacancy-atom exchange frequency, we have  $D_v = a^2\nu$  and  $\nu = \nu_0 e^{-E^m/(k_B T)}$  with  $E^m$  the migration energy,  $\nu_0$  the Debye frequency of aluminum,  $k_B$  Boltzmann's constant, and  $T$  the temperature (600 K). To compare between simulations and theory, it is convenient to define absorption efficiencies by renormalizing the sink strengths by the cavity concentration  $C_c$ :

$$\kappa = k^2/C_c. \quad (45)$$

We used Laplace and Wiedersich theoretical models [57] to compare the absorption efficiency estimated using our simulation model. The Laplace model is

described as

$$\kappa = 4\pi r_{cv} \quad (46)$$

where  $r_{cv} = r_{ca} + r_{va}$  is the sum of the cavity radius and the vacancy radius. The Wiedersich model depends on the sink concentration and is given by [57]

$$\kappa = 4\pi r_{cv} \frac{1 - \eta^3}{1 - \frac{9}{5}\eta + \eta^3 - \frac{1}{5}\eta^6}, \quad (47)$$

where  $\eta$  is  $r_{cv}/R$  and  $R$  is the average half distance between sinks (cavity). The distance  $R$  is calculated by  $R = \sqrt[3]{C_c}/2$  [26]. The computed absorption efficiencies are displayed in Fig. 13. The curve provided by Wiedersich model Eq.(47) have an almost perfect match with the simulation results. The constant value given by Laplace model Eq.(46) corresponds to absorbing efficiency in the limit of zero concentration of sinks. The absorption efficiency obtained in the simulations seem to converge to this value as concentration decreases. Another interesting feature obtained by the simulation is the negligible dependence of sink strengths on elastic interactions even for high concentration of the cavity, at least for the uniform initial distribution used in this work.

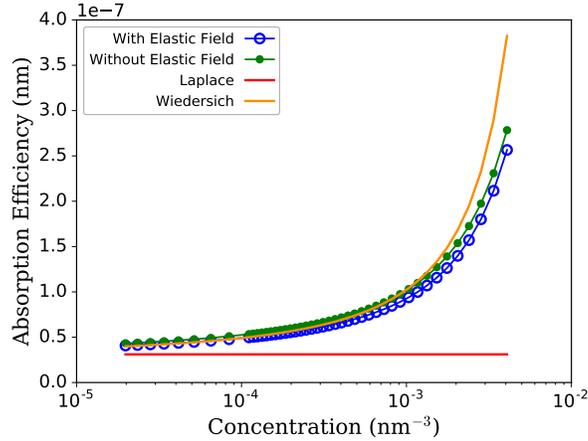


Figure 13: Comparison of absorption efficiencies calculated for the sink absorption model using MFPT from Eq.(23) and uniform initial distribution to those obtained from Laplace equation (46) and Wiedersich equation (47).

We note that the proposed algebraic approach to sink strength computations

possibly gives access to a lot of additional information, such as the total number of vacancy jumps and the distance traveled by the vacancy before absorption by the cavity, and also the dependence of the sink strengths on the initial probability distribution. The transient number of vacancy jumps is deduced from the residence time vector  $\vartheta^T = \boldsymbol{\pi}^T (\mathbf{A}^a)^{-1}$ , a quantity depending on the initial probability vector  $\boldsymbol{\pi}$ . The number of transient jumps from any state being equal to the product of the residence time and the jump frequency, the total number of jumps is obtained by summing over all transit and writes  $\sum_i \vartheta_i A_{ii}^a$ . Multiplying by  $a\sqrt{2}/2$ , the nearest neighbour distance between lattice site yields the distance traveled by the vacancy prior absorption. The later distance scaled by minimum distance separating the cavities, i.e. the period length, has been displayed in Fig. 14 for a range of cavity concentrations. We observe that the traveled distance increases much faster than the separating distance. The distance ratio follows as a parabolic law, entailing that the traveled distance grows as the cube of the separating distance.

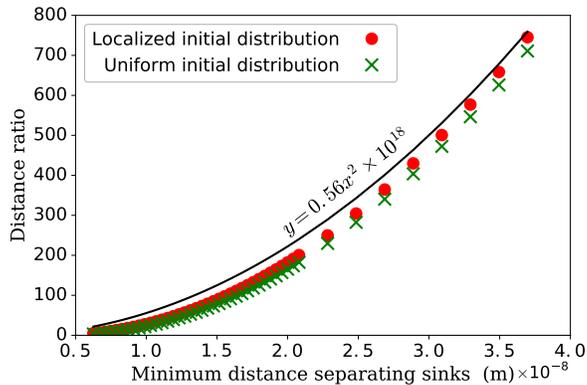


Figure 14: Distances traveled by the vacancy before absorption scaled by the minimum distance separating the sinks, for uniform and localized initial distributions, and plotted as a function of the separating distance.

We eventually quantify the anisotropy of the sink strengths associated with the localized initial distributions  $\mathbf{e}_j$ . Local sink strengths are defined by  $k^2(j) = 1/(\tau_j D_v)$  where  $\tau_j$  is the MFPT from site  $j$ . Their anisotropy, clearly visible

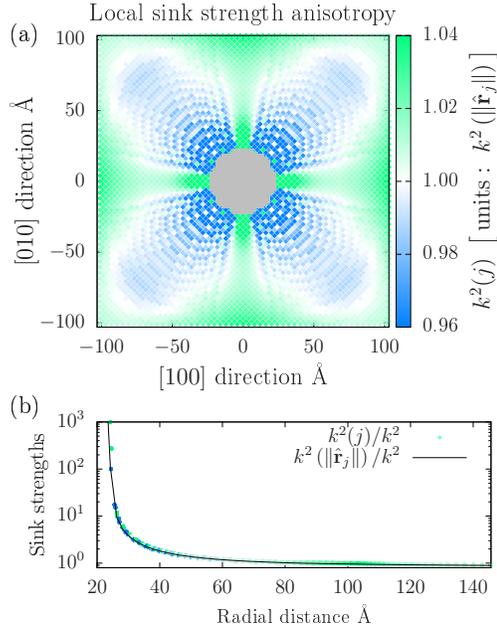


Figure 15: Using uniform distribution, (a) estimation of anisotropic sink strengths for the site  $j$ , and (b) evolution of the sink strengths parameter over a radial distance between the cavity and the vacancy.

in Fig. 15.a, is very moderate, in contrast to the radial dependence shown in Fig. 15.b.

This means that there is no need to account for elasticity to compute sink strengths of small cavities with respect to vacancies in aluminium under irradiation. Note that the size of the simulation box is restricted to  $3 \times 10^6$  sites due to memory constraints, and that the cavity sizes is also modest. We however expect a higher effect of the elastic field created by interstitial loops on the absorbing/emitting fluxes of point defects in aluminum [25]. The anisotropy of these fluxes may introduce a substantial angular dispersion of sink strengths which should ideally be taken into account in cluster dynamics simulations. This can be achieved by implementing the approach developed in a recent work [26]

in which the dispersion effect of the distances between the sinks is correctly accounted for in hybrid cluster dynamics simulations [58].

#### 4. Conclusions

In this work, we have developed several algorithms for characterizing a mobile defect's absorption kinetics by a periodic array of sinks, resorting to the theory of absorbing Markov chains. The sink concentration is determined by the cell size and the absorbing transition rate matrix is deduced from the transition rates between transient states. The initial probability vector is defined by the defect's initial distribution over the transient states. The goal is to predict the time evolution of the state probability vector of the defect until absorption. This entails computing the exponential of the constructed rate matrix, a task ideally accomplished via its full diagonalization. The standard eigensolvers that are used routinely to extract the entire eigenspectrum of low-dimensional matrices fail in high-dimensions due to memory constraints. Fortunately, in many diffusion problems, the involved transition matrices are extremely sparse. The usual approach for sparse high-dimensional matrices is to repeatedly extract portions of the eigenspectrum using iterative solvers based on deflation techniques. However, this still requires a substantial amount of computational time, restricting the range of applicability of this technique for simulating aging kinetics and microstructural evolution in materials science. We have succeeded in overcoming this issue by applying Krylov subspace projection techniques. This approach involves vector-matrix multiplications only and reduces the computational complexity by calculating the exponential of a much lower-dimensional transition rate matrix. Two algorithms have been developed, dubbed KSMP (Krylov Subspace Model Projection) and EKSMP (Eigenvalue and Krylov Subspace Model Projection). KSMP is based on constructing a Krylov basis starting from the initial probability vector and aiming at capturing its subsequent evolution. As the KSMP approach introduces a dependence on the initial vector, we have also developed and tested the additional EKSMP method to deflate the

Krylov subspace using the slowest eigenmodes. EKSMP and KSMP methods were implemented to study the absorption kinetics of a vacancy by a cavity in Aluminium. The correctness of the two algorithms was assessed by comparing the results obtained for a subset of times using the Restarted Krylov subspace projection (R-KSP) method as a reference: survival probabilities and first-passage distributions could be accurately reproduced using EKSMP and KSMP methods. Noticeably, an important simplification of the problem to solve stems from the fact that the diffusion process is reversible [31] entailing that all transition rate matrices can be symmetrized through diagonal similarity transformations. The reversibility condition is fulfilled in most applications involving the diffusion of defects even though these defects are created by an irreversible process like neutron, ion, or electron irradiation.

The crucial parameter controlling the convergence of KSMP and EKSMP methods is the dimension of the Krylov subspace. We found that the KSMP method yields accurate results with a Krylov basis whose dimension is five times the cubic root of the matrix dimension (the size of the three-dimensional lattice). Such a dimension for the Krylov basis (KB) makes it possible to capture long sequences of defect hops through the entire cell until absorption, and hence to account for the contribution of the QSD mode. We also observed that for our typical sink problem using the EKSMP method, the KB dimension is considerably reduced, by a factor of 10, even when only the QSD mode is included. This trend results from the fact that the QSD regime is reached very quickly and only involves the local diffusion of the defect. Besides, the extra cost associated with the QSD calculation being less or similar to the cost that is spared by reducing the Krylov subspace dimension, EKSMP method is more advantageous than KSMP method. This is especially true when more than one initial defect distribution is considered, a typical situation occurring in KMC and mean-field applications. Because the eigenvalues are often degenerate and eigenmodes occur in bundles, it was practically inconvenient and computationally expensive to include additional eigenmodes in EKSMP method. The working space dimension must be determined in the Krylov-Schur solver previously selected in our

applications for its superior performance compared to other iterative solvers. Overall, we show that it is unnecessary to extract several eigenmodes to characterize the absorbing kinetics fully.

Concerning sink strength calculations in which the initial defect distribution is homogeneous in space, the convergence with respect to the KB dimension is observed to be faster, as compared to situations with an initially localized vacancy. Overall, we show that Krylov subspace projection methods enable us to study the diffusion of a mobile defect around a sink in model system accounting for elastic dipole interactions and comprising up to million lattice sites. To compute the sink strengths at lower sink densities, the developed Krylov subspace projection technique should be implemented in combination with KMC simulations and conditioning techniques [59] to circumvent the curse of dimensionality and memory constraints. Finally, we point out that the proposed approach is based on the actual transition rate matrix on the crystal lattice. It may then straightforwardly be applied to investigate the anisotropic migration of interstitial clusters, whose diffusion mechanism mixes fast translations and slow rotations. The approach may allow to compute sink strengths relative any kind of defect clusters exhibiting mixed mobilities efficiently [27, 28].

### **Acknowledgements**

Fruitful discussions with Gilles Adjanor, Thomas Jourdan and Jose E. Roman are gratefully acknowledged.

### **Appendix A. Restarted Krylov Subspace Projection**

The restarted Krylov subspace algorithm proposed in ref.[44], generates Krylov basis of dimension  $\ell$ . In later step, that algorithm updates the approximation to  $f(\mathbf{A})\mathbf{b}$  and discards the basis vectors except the one which serves as an initial vector of the next Krylov subspace [60]. The following derivation is

to recall the restarted setup using two Lanczos decomposition

$$\mathbf{A}\mathbf{V}_\ell^1 = \mathbf{V}_\ell^1\mathbf{T}_\ell^1 + T_{\ell+1,\ell}^1\mathbf{v}_{\ell+1}^1\mathbf{e}_\ell^T \quad (\text{A.1})$$

$$\mathbf{A}\mathbf{V}_\ell^2 = \mathbf{V}_\ell^2\mathbf{T}_\ell^2 + T_{\ell+1,\ell}^2\mathbf{v}_{\ell+1}^2\mathbf{e}_\ell^T \quad (\text{A.2})$$

where  $\mathbf{V}^1$  and  $\mathbf{V}^2$  are the orthonormal bases of  $\mathcal{K}_\ell(\mathbf{A}, \mathbf{v}_1)$  and  $\mathcal{K}_\ell(\mathbf{A}, \mathbf{v}_{\ell+1})$ .  $\mathbf{T}_\ell^1$  and  $\mathbf{T}_\ell^2$  are two tridiagonal matrices.  $\mathbf{e}_\ell^T$  denotes the  $\ell$ th unit coordinate vector  $\in \mathbb{R}^\ell$ . Together the columns of  $\mathbf{W}_{2\ell} := [\mathbf{V}_\ell^1, \mathbf{V}_\ell^2]$  forms a basis of  $\mathcal{K}_{2\ell}(\mathbf{A}, \mathbf{b})$ . On combining the two Lanczos decomposition Eq.(A.1) and Eq.(A.2) to Lanczos-like decomposition we get,

$$\mathbf{A}\mathbf{W}_{2\ell} = \mathbf{W}_{2\ell}\mathbf{T}_{2\ell} + T_{\ell+1,\ell}^2\mathbf{v}_{\ell+1}^2\mathbf{e}_{2\ell}^T \quad (\text{A.3})$$

where  $\mathbf{T}_{2\ell}$  is the tridiagonal block Matrix represented as,

$$\mathbf{T}_{2\ell} := \begin{bmatrix} \mathbf{T}_\ell^1 & \mathcal{O} \\ T_{\ell+1,\ell}^1\mathbf{e}_1\mathbf{e}_\ell^T & \mathbf{T}_\ell^2 \end{bmatrix}. \quad (\text{A.4})$$

The restarted method of Krylov subspace approximation associated to Eq.(A.3) is given as,

$$\mathbf{f}_{2\ell} = \beta\mathbf{W}_{2\ell}f(\mathbf{T}_{2\ell})\mathbf{e}_1. \quad (\text{A.5})$$

The  $f(\mathbf{T}_{2\ell})$  term exhibits the following block lower triangular structure,

$$f(\mathbf{T}_{2\ell}) = \begin{bmatrix} f(\mathbf{T}_\ell^1) & \mathcal{O} \\ \mathbf{X}_{2,1} & f(\mathbf{T}_\ell^2) \end{bmatrix}. \quad (\text{A.6})$$

Hence, the approximation Eq.(A.5) has the form,

$$\mathbf{f}_{2\ell} = \beta\mathbf{V}_\ell^1f(\mathbf{T}_\ell^1)\mathbf{e}_1 + \beta\mathbf{V}_\ell^2\mathbf{X}_{2,1}\mathbf{e}_1 \quad (\text{A.7})$$

where the first term of Eq.(A.7) is evaluated using Arnoldi approximation for the basis  $\mathcal{K}_\ell(\mathbf{A}, \mathbf{b})$ . Once the  $\mathbf{X}_{2,1}\mathbf{e}_1$  is estimated, the basis vectors of  $\mathbf{V}_\ell^1$  are discarded and Eq.(A.7) yields the basis of restarting method by updating the Arnoldi approximation. The approximation after  $m$  restart cycles is given as

$$\mathbf{f}^m = \beta\mathbf{W}_{m\ell}f(\mathbf{T}_{m\ell})\mathbf{e}_1 = \mathbf{f}^{(m-1)} + \beta\mathbf{V}_\ell^m[f(\mathbf{T}_{m\ell})\mathbf{e}_1]_{(m-1)\ell+1:m\ell} \quad (\text{A.8})$$

where the subscript of the last term in Eq.(A.8) represents the vector with the last  $\ell$  components of  $f(\mathbf{T}_{m\ell})\mathbf{e}_1$  [44].

## Appendix B. Cholesky preconditioning

CPU times for performing ESMP or EKSMP simulations are compiled in Tables B.2, B.3 and B.4. From data reported in Table B.2, we observe that Cholesky preconditioning (CP) should be performed whenever possible because it reduces the overall CPU times and improves the convergence of the KS solver. Eigenvalues being pooled in bundles for symmetry reasons, eigenvectors appears simultaneously in the extraction algorithm. Handling the eigenvalue degeneracy is facilitated by the inverted iterations within CP. CP is limited in memory because the computed Cholesky factor is a denser matrix. The largest system CP can solve contains 217245 transient state, as shown in table B.3. For the larger systems reported in Table B.4, the QSD eigenvector was successively computed using KS solver without preconditioning up to 905681 transient states.

Input		Output				
CP	NEV	CP Time (s)	EPS Time (s)	Converged Eigenpairs	NCV	Result
Disabled	1	-	$8.93 \cdot 10^1$	1	16	C
	5	-	$1.23 \cdot 10^2$	1	20	D
	10	-	$1.30 \cdot 10^2$	2	25	D
	30	-	$1.91 \cdot 10^2$	21	60	D
	40	-	$2.25 \cdot 10^2$	31	80	D
	50	-	$2.35 \cdot 10^2$	53	100	C
	60	-	$1.91 \cdot 10^2$	60	120	C
	70	-	$1.94 \cdot 10^2$	70	140	C
	80	-	$2.34 \cdot 10^2$	82	160	C
	90	-	$2.45 \cdot 10^2$	90	180	C
	100	-	$2.89 \cdot 10^2$	102	200	C
Enabled	1	$1.58 \cdot 10^0$	$1.55 \cdot 10^1$	1	1	C
	5	$5.53 \cdot 10^0$	$1.98 \cdot 10^1$	7	7	C
	10	$5.92 \cdot 10^0$	$1.99 \cdot 10^1$	11	11	C
	30	$1.54 \cdot 10^1$	$3.13 \cdot 10^1$	37	60	C
	40	$1.94 \cdot 10^1$	$3.55 \cdot 10^1$	42	80	C
	50	$2.06 \cdot 10^1$	$3.52 \cdot 10^1$	60	100	C
	60	$2.20 \cdot 10^1$	$3.70 \cdot 10^1$	64	120	C
	70	$2.33 \cdot 10^1$	$3.99 \cdot 10^1$	73	140	C
	80	$2.66 \cdot 10^1$	$4.37 \cdot 10^1$	81	160	C
	90	$3.05 \cdot 10^1$	$4.71 \cdot 10^1$	90	180	C
	100	$3.75 \cdot 10^1$	$5.69 \cdot 10^1$	114	200	C
	3500	$1.52 \cdot 10^3$	$4.73 \cdot 10^3$	3539	4000	C

Table B.2: CPU time taken by ESMP method to extract the indicated number of eigenvalues (NEV) for the vacancy absorption model. The simulations are performed using a single core of an Intel i5-8400H processor (running at 2.5GHz). Parameter NCV represents the maximum dimension of the working subspace to be used by the solver. In inputs, Cholesky preconditioning (CP) may be enabled or disabled. In the outputs, C and D of the result column denote whether simulations have converged or diverged, respectively.

Cell size (Å)	Number of Transient States ( $N$ )	Time (s)
66.66	14141	$3.55 \cdot 10^0$
74.74	21085	$7.83 \cdot 10^0$
86.16	34801	$2.13 \cdot 10^1$
98.98	53053	$4.36 \cdot 10^1$
107.06	68061	$7.06 \cdot 10^1$
127.26	116921	$2.36 \cdot 10^2$
139.38	154973	$4.83 \cdot 10^2$
147.46	184381	$6.31 \cdot 10^2$
151.5	200369	$7.91 \cdot 10^2$
159.58	217245	$9.80 \cdot 10^2$

Table B.3: CPU time taken by ESMP to extract 10 eigenvalues for the indicated number of system size ( $N$ ) using KS solver and Cholesky preconditioning from MUMPS.

Cell size (Å)	Number of Transient States ( $N$ )	Time (s)
74.74	21085	$4.42 \cdot 10^1$
86.86	34801	$1.11 \cdot 10^2$
98.98	53053	$2.33 \cdot 10^2$
107.06	68061	$4.10 \cdot 10^2$
119.18	95313	$5.23 \cdot 10^2$
127.26	116921	$7.47 \cdot 10^2$
139.38	154973	$1.20 \cdot 10^3$
147.46	184381	$1.57 \cdot 10^3$
151.50	200369	$1.80 \cdot 10^3$
159.58	235033	$2.62 \cdot 10^3$
167.66	273441	$3.26 \cdot 10^3$
187.86	387101	$5.66 \cdot 10^3$
208.06	497757	$9.33 \cdot 10^3$
248.46	905681	$2.38 \cdot 10^4$

Table B.4: CPU time to evaluate survival probability distributions for the indicated number of system size ( $N$ ) using EKSMP method with  $(k, \ell) = (1, 50)$  and KS solver without Cholesky preconditioning.

- [1] N.-V. Buchete, G. Hummer, Coarse Master Equations for Peptide Folding Dynamics, *The Journal of Physical Chemistry B* 112 (2008) 6057–6069. Publisher: American Chemical Society.
- [2] T. Opperstrup, V. V. Bulatov, A. Donev, M. H. Kalos, G. H. Gilmer, B. Sadigh, First-passage kinetic Monte Carlo method, *Physical Review E* 80 (2009) 066701. Publisher: American Physical Society.
- [3] D. Iracane, P. Chaix, A. Alamo, Jules Horowitz Reactor: a high performance material testing reactor, *Comptes Rendus Physique* 9 (2008) 445–456.
- [4] R. G. Carter, N. Soneda, K. Dohi, J. M. Hyde, C. A. English, W. L. Server, Microstructural characterization of irradiation-induced Cu-enriched clusters in reactor pressure vessel steels, *Journal of Nuclear Materials* 298 (2001) 211–224.
- [5] W. J. Phythian, C. A. English, Microstructural evolution in reactor pressure vessel steels, *Journal of Nuclear Materials* 205 (1993) 162 – 177.
- [6] J. T. Buswell, W. J. Phythian, R. J. McElroy, S. Dumbill, P. H. N. Ray, J. Mace, R. N. Sinclair, Irradiation-induced microstructural changes, and hardening mechanisms, in model PWR reactor pressure vessel steels, *Journal of Nuclear Materials* 225 (1995) 196–214.
- [7] T. J. Frankcombe, S. C. Smith, Numerical solution methods for large, difficult kinetic master equations, *Theoretical Chemistry Accounts* 124 (2009) 303.
- [8] A. Bortz, M. Kalos, J. Lebowitz, A new algorithm for monte carlo simulation of ising spin systems, *Journal of Computational Physics* 17 (1975) 10 – 18.
- [9] R. Ngayam-Happy, C. S. Becquart, C. Domain, L. Malerba, Formation and evolution of MnNi clusters in neutron irradiated dilute Fe alloys modelled

- by a first principle-based AKMC method, *Journal of Nuclear Materials* 426 (2012) 198–207.
- [10] E. Martinez, P. Monasterio, J. Marian, Billion-atom synchronous parallel kinetic monte carlo simulations of critical 3d ising systems, *Journal of Computational Physics* 230 (2011) 1359 – 1369.
- [11] G. Nandipati, A. Kara, S. I. Shah, T. S. Rahman, Off-lattice pattern recognition scheme for kinetic monte carlo simulations, *Journal of Computational Physics* 231 (2012) 3548 – 3560.
- [12] M. Trochet, N. Mousseau, L. K. Béland, G. Henkelman, Off-lattice kinetic monte carlo methods, *Handbook of Materials Modeling: Methods: Theory and Modeling* (2020) 715–743.
- [13] C. Domain, C. S. Becquart, Object kinetic monte carlo (okmc): A coarse-grained approach to radiation damage, *Handbook of Materials Modeling: Methods: Theory and Modeling* (2020) 1287–1312.
- [14] J.-M. Lanore, Simulation de l'évolution des défauts dans un réseau par le méthode de monte-carlo, *Radiation Effects* 22 (1974) 153–162.
- [15] T. L. Hoang, J. Marian, V. V. Bulatov, P. Hosemann, Computationally-efficient stochastic cluster dynamics method for modeling damage accumulation in irradiated materials, *Journal of Computational Physics* 300 (2015) 254–268.
- [16] T. Opplestrup, V. V. Bulatov, G. H. Gilmer, M. H. Kalos, B. Sadigh, First-Passage Monte Carlo Algorithm: Diffusion without All the Hops, *Physical Review Letters* 97 (2006) 230602.
- [17] A. Donev, V. V. Bulatov, T. Oppelstrup, G. H. Gilmer, B. Sadigh, M. H. Kalos, A first-passage kinetic monte carlo algorithm for complex diffusionreaction systems, *Journal of Computational Physics* 229 (2010) 3214 – 3236.

- [18] A. F. Voter, Introduction to the kinetic monte carlo method, in: Radiation effects in solids, Springer, 2007, pp. 1–23.
- [19] C. Daniels, P. Bellon, Hybrid kinetic monte carlo algorithm for strongly trapping alloy systems, Computational Materials Science 173 (2020) 109386.
- [20] M. A. Novotny, Monte Carlo Algorithms with Absorbing Markov Chains: Fast Local Algorithms for Slow Dynamics, Physical Review Letters 74 (1995) 1–5. Publisher: American Physical Society.
- [21] B. Puchala, M. L. Falk, K. Garikipati, An energy basin finding algorithm for kinetic Monte Carlo acceleration, The Journal of Chemical Physics 132 (2010) 134104.
- [22] M. Athnes, V. V. Bulatov, Path Factorization Approach to Stochastic Simulations, Physical Review Letters 113 (2014) 230601. Publisher: American Physical Society.
- [23] G. Nandipati, Y. Shim, J. G. Amar, First-passage time approach to kinetic Monte Carlo simulations of metal (100) growth, Physical Review B 81 (2010) 235415.
- [24] S. Redner, A guide to first-passage processes, Cambridge University Press, 2001.
- [25] D. Carpentier, T. Jourdan, Y. Le Bouar, M. C. Marinica, Effect of saddle point anisotropy of point defects on their absorption by dislocations and cavities, Acta Materialia 136 (2017) 323–334.
- [26] D. Carpentier, T. Jourdan, P. Terrier, M. Athnes, Y. Le Bouar, Effect of sink strength dispersion on cluster size distributions simulated by cluster dynamics, Journal of Nuclear Materials 533 (2020) 152068.
- [27] G. Adjanor, Complete characterization of sink-strengths for 1d to 3d mobilities of defect clusters i. extension to diffusion anisotropy analog cases, arXiv preprint arXiv:1808.10362 (2018).

- [28] G. Adjanor, Complete characterization of sink-strengths for 1d to 3d mobilities of defect clusters. ii. bridging between limiting cases with effective sink-strengths calculations, arXiv preprint arXiv:1808.10715 (2018).
- [29] T. Jourdan, G. Bencteux, G. Adjanor, Efficient simulation of kinetics of radiation induced defects: A cluster dynamics approach, *Journal of Nuclear Materials* 444 (2014) 298 – 313.
- [30] L. Malerba, C. S. Becquart, C. Domain, Object kinetic Monte Carlo study of sink strengths, *Journal of Nuclear Materials* 360 (2007) 159–169.
- [31] M. Athnes, S. Kaur, G. Adjanor, T. Vanacker, T. Jourdan, Elastodiffusion and cluster mobilities using kinetic Monte Carlo simulations: Fast first-passage algorithms for reversible diffusion processes, *Physical Review Materials* 3 (2019) 103802.
- [32] T. D. Swinburne, D. Kannan, D. J. Sharpe, D. J. Wales, Rare events and first passage time statistics from the energy landscape, *The Journal of Chemical Physics* 153 (2020) 134115. Publisher: American Institute of Physics.
- [33] M. R. Hestenes, E. Stiefel, et al., *Methods of conjugate gradients for solving linear systems*, volume 49, NBS Washington, DC, 1952.
- [34] S.-C. T. Choi, C. C. Paige, M. A. Saunders, MINRES-QLP: A Krylov Subspace Method for Indefinite or Singular Symmetric Systems, *SIAM Journal on Scientific Computing* 33 (2011) 1810–1836. Publisher: Society for Industrial and Applied Mathematics.
- [35] Y. Saad, *Iterative methods for sparse linear systems*, SIAM, 2003.
- [36] A. Amritkar, E. de Sturler, K. wirydownicz, D. Tafti, K. Ahuja, Recycling Krylov subspaces for CFD applications and a new hybrid recycling solver, *Journal of Computational Physics* 303 (2015) 222–237.

- [37] G. W. Stewart, A krylov–schur algorithm for large eigenproblems, *SIAM Journal on Matrix Analysis and Applications* 23 (2002) 601–614.
- [38] P. Arbenz, *Lecture Notes on Solving Large Scale Eigenvalue Problems*, Technical Report, Computer Science Department, ETH Zürich, 2016.
- [39] C. L. Bris, T. Lelivre, M. Luskin, D. Perez, A mathematical formalization of the parallel replica dynamics 18 (2012) 119–146.
- [40] G. Di Ges, T. Lelivre, D. Le Peutrec, B. Nectoux, Jump Markov models and transition state theory: the quasi-stationary distribution approach, *Faraday Discussions* 195 (2016) 469–495.
- [41] J. E. Roman, C. Campos, E. Romero, A. Tomas, *SLEPc Users Manual*, Technical Report DSIC-II/24/02 - Revision 3.11, D. Sistemes Informàtics i Computació, Universitat Politècnica de València, 2019.
- [42] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, V. Eijkhout, W. D. Gropp, D. Karpeyev, D. Kaushik, M. G. Knepley, D. A. May, L. C. McInnes, R. T. Mills, T. Munson, K. Rupp, P. Sanan, B. F. Smith, S. Zampini, H. Zhang, H. Zhang, *PETSc Users Manual*, Technical Report ANL-95/11 - Revision 3.11, Argonne National Laboratory, 2019.
- [43] S. Balay, S. Abhyankar, M. F. Adams, J. Brown, P. Brune, K. Buschelman, L. Dalcin, A. Dener, V. Eijkhout, W. D. Gropp, D. Karpeyev, D. Kaushik, M. G. Knepley, D. A. May, L. C. McInnes, R. T. Mills, T. Munson, K. Rupp, P. Sanan, B. F. Smith, S. Zampini, H. Zhang, H. Zhang, *PETSc Web page*, <https://www.mcs.anl.gov/petsc>, 2019. URL: <https://www.mcs.anl.gov/petsc>.
- [44] M. Eiermann, O. G. Ernst, A Restarted Krylov Subspace Method for the Evaluation of Matrix Functions, *SIAM Journal on Numerical Analysis* 44 (2006) 2481–2504. Publisher: Society for Industrial and Applied Mathematics.

- [45] P. R. Amestoy, I. S. Duff, J.-Y. L'Excellent, J. Koster, A Fully Asynchronous Multifrontal Solver Using Distributed Dynamic Scheduling, *SIAM Journal on Matrix Analysis and Applications* 23 (2001) 15–41. Publisher: Society for Industrial and Applied Mathematics.
- [46] I. C. F. Ipsen, C. D. Meyer, The idea behind krylov methods, *The American Mathematical Monthly* 105 (1998) 889–899.
- [47] A. C. Antoulas, *Approximation of large-scale dynamical systems*, SIAM, 2005.
- [48] N. J. Higham, A. H. Al-Mohy, Computing matrix functions, *Acta Numerica* 19 (2010) 159–208. Publisher: Cambridge University Press.
- [49] O. Coulaud, L. Giraud, P. Ramet, X. Vasseur, Deflation and augmentation techniques in Krylov subspace methods for the solution of linear systems, *arXiv:1303.5692 [math]* (2013). ArXiv: 1303.5692.
- [50] M. Gutknecht, Spectral deflation in krylov solvers: A theory of coordinate space based methods, *ETNA. Electronic Transactions on Numerical Analysis [electronic only]* 39 (2012).
- [51] A. Gaul, M. H. Gutknecht, J. Liesen, R. Nabben, A Framework for Deflated and Augmented Krylov Subspace Methods, *SIAM Journal on Matrix Analysis and Applications* 34 (2013) 495–518. Publisher: Society for Industrial and Applied Mathematics.
- [52] V. Hernandez, J. E. Roman, V. Vidal, SLEPc: A scalable and flexible toolkit for the solution of eigenvalue problems, *ACM Trans. Math. Software* 31 (2005,) 351–362.
- [53] V. Hernandez, J. E. Roman, A. Tomas, V. Vidal, A survey of software for sparse eigenvalue problems, Technical Report STR-6, Universitat Politècnica de València, 2009. Available at <https://slepc.upv.es>.

- [54] S. Balay, W. D. Gropp, L. C. McInnes, B. F. Smith, Efficient management of parallelism in object oriented numerical software libraries, in: E. Arge, A. M. Bruaset, H. P. Langtangen (Eds.), *Modern Software Tools in Scientific Computing*, Birkhäuser Press, 1997, pp. 163–202.
- [55] C. N. Haddad, *Cholesky factorization*, Springer Science & Business Media, 2009, pp. 374–377.
- [56] C. A. Floudas, P. M. Pardalos, *Encyclopedia of optimization*, Springer Science & Business Media, 2008.
- [57] H. Wiedersich, On the theory of void formation during irradiation, *Radiation Effects* 12 (1972) 111–125.
- [58] P. Terrier, M. Athnes, T. Jourdan, G. Adjanor, G. Stoltz, Cluster dynamics modelling of materials: A new hybrid deterministic/stochastic coupling approach, *Journal of Computational Physics* 350 (2017) 280–295.
- [59] M. Athènes, Conditioning and enhanced sampling schemes for simulating thermodynamic and kinetic properties of condensed matter, *Habilitation à diriger des recherches*, Université Paris Saclay ; Université paris sud, 2018. URL: <https://hal-cea.archives-ouvertes.fr/tel-01851686>.
- [60] M. Afanasjew, M. Eiermann, O. G. Ernst, S. Gttel, Implementation of a restarted Krylov subspace method for the evaluation of matrix functions, *Linear Algebra and its Applications* 429 (2008) 2293–2314.