A non-iterative local remeshing approach for simulating moving boundary transient diffusion problems

Soheil Soghrati\textsuperscript{a,b,*}, Yuhao Chen\textsuperscript{a}, Weijie Mai\textsuperscript{b}

\textsuperscript{a}Department of Mechanical and Aerospace Engineering, The Ohio State University, Columbus, OH, USA
\textsuperscript{b}Department of Materials Science and Engineering, The Ohio State University, Columbus, OH, USA

Abstract

A computational framework relying on a non-iterative mesh generation algorithm is introduced for modeling moving boundary transient diffusion problems, with a special focus on its application for simulating corrosion phenomena. In this approach, the Conforming to Interface Structured Adaptive Mesh Refinement (CISAMR) technique is employed to adapt the finite element mesh to the evolving morphology of the problem. CISAMR combines customized $h$-adaptivity, $r$-adaptivity, and sub-triangulation algorithms to transform a simple structured mesh into a high quality hybrid conforming mesh composed of rectangular and triangular elements. A key advantage of this method for modeling moving boundary problems is that only the elements located along the moving boundary must be modified to regenerate a conforming mesh at each step. This feature not only facilitates the remeshing process but also reduces the error associated with projecting nodal values of the solution to the new mesh. After a convergence study, we verify the accuracy of CISAMR for modeling pitting corrosion problems by comparing results with analytical solutions and phase field simulations. Additional examples are also provided to show the application of CISAMR for simulating corrosion problems with intricate morphologies.

\*Corresponding author: Departments of Mechanical and Aerospace Engineering & Materials Science and Engineering, The Ohio State University, Columbus, OH, USA.

Email address: soghrati.1@osu.edu (Soheil Soghrati)

Preprint submitted to Elsevier

November 13, 2017
1. Introduction

Several physical phenomena including the phase transformation are formulated as transient moving boundary problems governed by the diffusion law [1, 2], which are also referred to as Stefan problems [3, 4]. A major challenge toward simulating such problems using the finite element method (FEM) is the evolution of the domain morphology, which necessitates continuous updating of the mesh structure to conform to the moving boundary [5, 6]. Varying robust algorithms can be employed to generate finite element (FE) meshes with proper element aspect ratios and negligible discretization error, among which we can mention the Delaunay triangulation [7, 8, 9], advancing front [10, 11], quadtree/octree-based techniques [12, 13, 14], marching cubes [15], and the dual contouring method [16, 17, 18]. However, the computational cost associated with reconstructing a new conforming mesh at each time step for simulating moving boundary problems using these methods could be overwhelming. This is in part due to the iterative/optimization phase involved in such algorithms to improve the aspect ratios of elements. For example, a Laplacian smoothing is often used in quadtree-based algorithms to create elements with proper aspect ratios by iteratively relocating the mesh nodes [13, 19]. Another challenge associated with the remeshing process is the requirement to project the solution between the nodes of the old (deformed) and new (reconstructed) meshes that coexist at each time step [20]. This nodal projection not only imposes an additional computational burden but also deteriorates the accuracy and convergence rate by undermining the super-convergence feature of FEM [21].

Alternative techniques such as the Arbitrary Lagrangian-Eulerian (ALE) method [22, 23, 24, 25] can be implemented to evolve a conforming mesh during an FE simulation without remeshing the entire domain at each time step. In the ALE method, after evaluating the updated morphology of the domain at each time step, techniques such as the relocation of mesh nodes [26, 27] and edge swapping [28] are employed to improve the quality of elements (i.e., their shape and aspect ratio) [29]. For problems with significant geometrical changes, a combination of smoothing algorithms and modifying the interface velocity has been used to prevent the element tangling issue [30].
all these treatments, remeshing may still be required after a certain number of time steps for problems with intricate evolving geometries and/or those undergoing large deformations [26, 31]. Automatic mesh moving techniques, often combined with interface-tracking and space-time methods [32, 33], have also been introduced to limit the burden of remeshing for modeling moving boundary problems [34, 35, 36, 37, 38]. Similar to the ALE method, iterative nodal repositioning, edge/face swap, and mesh optimization techniques have been used in such methods to avoid the creation of tangling elements and improve their aspect ratios [39, 40, 41]. Recently, Gawlik and Lew [42] have introduced a robust technique for modeling 2D moving boundary problems that employs an iterative relaxation algorithm to adapt a stationary background mesh (universal mesh) to the evolving interface geometry [43, 44].

To obviate the challenges associated with the remeshing process, one can implement meshfree techniques such as the smoothed particle hydrodynamics [45, 46], element-free Galerkin method [47, 48], exponential basis functions meshfree technique [49, 50], and the Green’s discrete transformation method [51]. The idea of making the solution field independent of the mesh structure can also be incorporated in the FE formulation by appropriate enrichment strategies [52, 53, 54]. The eXtended/Generalized FEM (X/GFEM)[55, 56, 57, 58, 59] is one of the most popular techniques in this category that relies on enrichment functions constructed using the partition of unity method to reconstruct strong/weak discontinuities in nonconforming elements. This method has successfully been implemented for modeling a variety of moving boundary problems, e.g., [60, 61, 62, 63, 64, 65, 66, 67]. Compared to remeshing and ALE techniques, X/GFEM provides a major advantage by allowing the use of a stationary nonconforming mesh for modeling problems with evolving morphologies. XFEM has also been combined with the grid-based particle method [68] to tracks interface motion with meshless particles and handle topological changes such as boundary merging [69]. It is worth mentioning that additional treatments might be required in enriched methods such as X/GFEM to resolve implementation issues such as the high condition number of the stiffness matrix [70, 71] and enforcing Dirichlet boundary conditions [72, 73].

Another approach that can eliminate the need to create conforming meshes during the FE simulation of moving boundary problems is the phase field method [74, 75]. In this method, a diffuse interface model is employed to approximate the strong discontinuity across the interface as a continuously varying function with a pre-defined thickness [76]. The phase field method
has been employed to simulate a variety of moving boundary problems, such as the solidification [77, 78], dislocation interactions [79, 80], and corrosion [81, 82, 83]. While eliminating the need to generate conforming meshes, the phase field method requires a highly refined (nonconforming) mesh in the vicinity of the diffuse interface to accurately approximate the sharp gradients of the fields in this region. Further, an additional phase field variable must be incorporated in the problem formulation to implicitly track the interface location [76], which leads to a higher computational cost compared to sharp interface models. The peridynamics (PD) model introduced in [84, 85] is an alternative diffuse interface model that can be employed for simulating moving boundary problems. In the PD model, the domain is discretized using a structured grid, where each node interacts with its neighboring nodes within a certain distance. The phase transformation is achieved by monitoring the concentration associated with each node, which unlike the phase field method does not use an additional variable to track the interface location.

In this manuscript, we introduce a computational framework relying on a non-iterative mesh generation algorithm named Conforming to Interface Structured Adaptive Mesh Refinement (CISAMR) [86] for simulating moving boundary transient diffusion problems. The concept behind this approach is similar to the universal meshing method [42], as we regenerate the FE model after a few time steps by transforming a background mesh into a conforming mesh. However, CISAMR performs this transformation using a non-iterative algorithm (versus the iterative relaxation approach in universal remeshing) that only affects the locations of nodes of background elements intersecting with the moving interface. This algorithm generates a hybrid conforming mesh composed of quadrilateral and triangular elements with proper aspect ratios. An $h$-adaptive refinement phase is incorporated in the CISAMR algorithm that enables reducing the geometric discretization error and approximating sharp gradients along the moving interface more accurately. This technique can handle moving boundary problems undergoing large changes in the domain morphology without compromising the mesh quality. Among a variety of transient diffusion problems with evolving geometries, in this work we focus on the application of CISAMR for simulating corrosion phenomena and in particular the pitting corrosion [87, 88]. It must be noted that although the focus of the current manuscript is on modeling 2D problems, there is no inherent limitation for expanding the CISAMR non-iterative algorithm to 3D.

The remainder of this article is structured as follows. In Section 2 we
introduce strong and weak forms of transient diffusion governing equations, together with the special considerations required for modeling the pitting corrosion phenomenon \cite{89, 90}. A brief overview of the CISAMR algorithm and the algorithmic aspects pertaining to its application for modeling moving boundary problems are presented in Section 3. Several numerical examples, including a detailed convergence study, are presented in Section 4 to verify the accuracy of CISAMR simulations through comparison with analytical results and phase field simulations. We also show the application of this method for simulating several geometrically elaborate corrosion problems, such as the electropolishing and pitting corrosion phenomena. Final concluding remarks are presented in Section 5.

2. Governing equations

Consider an open domain $\Omega = \Omega_s \cup \Omega_f$ consisting of solid $\Omega_s$ and fluid $\Omega_f$ phases, with the boundary $\Omega \setminus \Omega = \Gamma$ and the outward unit normal vector $\mathbf{n}$. The domain boundary is composed of three non-overlapping partitions $\Gamma_D$, $\Gamma_N$, and $\Gamma_R$, corresponding to the Dirichlet, Neumann, and Robin boundary conditions, respectively. Also, $\Gamma_S$ refers to the interface between the fluid and the solid phases, which moves with velocity $\mathbf{v}$ during the transient diffusion phenomenon. The strong form of governing equations describing this process are given by: Find the transient field $c(\mathbf{x}, t)$ such that

$$
\frac{\partial c(\mathbf{x}, t)}{\partial t} = D \nabla^2 c(\mathbf{x}, t) \quad \text{in } \Omega_f \\
c(\mathbf{x}, 0) = c_0 \quad \text{in } \Omega_f \\
c(\mathbf{x}, t) = c_{\text{solid}} \quad \text{in } \Omega_s \\
c(\mathbf{x}, t) = \bar{c} \quad \text{on } \Gamma_D \\
\nabla c(\mathbf{x}, t) \cdot \mathbf{n} = \bar{\mathbf{q}} \quad \text{on } \Gamma_N \\
-D \nabla c(\mathbf{x}, t) \cdot \mathbf{n} + h c(\mathbf{x}, t) = g \quad \text{on } \Gamma_R,
$$

where $D$ is the diffusivity, $c_0$ is the initial distribution of the field, $\bar{c}$ is the prescribed value of the field along $\Gamma_D$, $\bar{\mathbf{q}}$ is the applied flux along $\Gamma_N$, and $h$ and $g$ are problem-specific constants.

While (1) can be employed for modeling a variety of moving boundary problems governed by the diffusion law, in the current manuscript we focus on the pitting corrosion phenomenon as the physical application. Pitting corrosion is the localized degradation of a metal due to the partial breakdown of the passive film protecting its surface against a corrosive environment.
Although high strength alloys such as stainless steel and 7xxx-series aluminum alloys are resistant to uniform corrosion, they are susceptible to pitting corrosion in environments with aggressive anions such as chloride. Pitting corrosion is devastating not only due to the mass loss caused by reverse metallurgical processes but also because growing pits induce significant stress concentrations that accelerates the crack nucleation and reduces the fatigue life [92]. Figure 1 schematically illustrates the domain of a pitting corrosion problem, consisting of the metal and electrolyte phases, together with the initial and boundary conditions assigned to each phase.

For pitting corrosion problems, the concentration of dissolved ions in the electrolyte solution and the velocity of the moving pit boundary can be described using the activation-diffusion-controlled pitting corrosion model presented in [90]. In this model, the evolution of the corroding interface $\Gamma_S$ is characterized using the Rankine-Hugoniot equation, which can be described as [89]

$$-D \nabla c(x, t) \cdot n + [c_{\text{solid}} - c(x, t)]v_n = 0 \text{ on } \Gamma_S,$$

where $v_n = v \cdot n$ is the normal component of the interface velocity. The numerical implementation of this condition depends on the relative rates of the metal dissolution and the ions diffusion in the electrolyte. If the metal dissolution rate is relatively fast, the concentration of dissolved ions gradually increases and would eventually reach the saturation concentration $c_{\text{sat}}$. At this stage, dissolved metal ions react with Cl$^{-}$ ions in the electrolyte.
and precipitate a thin salt film along the pit interface, which does not allow further increase of the concentration. After the formation of the salt film, the diffusion law governs the release of the ions from the film into the electrolyte and thereby controls the pit growth rate, which is often referred to as the diffusion-controlled pitting corrosion phenomenon. For modeling this process, a Dirichlet boundary condition of $\bar{c} = c_{\text{sat}}$ is imposed along the pit interface $\Gamma_S$. The interface velocity is then computed by rearranging (2) as a function of the concentration gradient, i.e.,

$$v_n = \frac{D \nabla c(x, t) \cdot \mathbf{n}}{c_{\text{solid}} - c_{\text{sat}}}, \quad \text{if } c(x, t)|_{\Gamma_S} = c_{\text{sat}}. \quad (3)$$

Unlike the diffusion-controlled pitting corrosion, before the formation of the salt film the interface velocity is controlled by the overpotential $\eta$. At this stage, the pit interface velocity $v_n$ is evaluated based on the Tafel electrochemical kinetics and the Faraday’s second law, which is given by [89, 91]

$$v_n = \frac{i_0 e^{\eta / nF}}{nF c_{\text{solid}}}, \quad \text{if } c(x, t)|_{\Gamma_S} < c_{\text{sat}}, \quad (4)$$

where $i_0$ is the exchange current density, $b_a$ is the anodic Tafel slope, $F$ is the Faraday’s constant, and $n$ is the average charge number. The Robin boundary condition along $\Gamma_S$ during the activation-controlled pitting is obtained by substituting (4) in the Rankine-Hugoniot condition given in (2).

After deriving the weak form of the governing equations in (1) and using the backward Euler scheme for the time discretization, the solution field $c_{n+1}^h$ at time $t_{n+1}$ can be approximated as

$$\int_{\Omega_f} \frac{1}{\Delta t} wc_{n+1}^h \, d\Omega + \int_{\Omega_f} D \nabla w \cdot \nabla c_{n+1}^h \, d\Omega - \int_{\Gamma_D} wc_{n+1}^h v_n d\Gamma =$$

$$\int_{\Omega_f} \frac{1}{\Delta t} wc_n^h \, d\Omega + \int_{\Gamma_N} w \tilde{q} \, d\Gamma + \int_{\Gamma_D} L w v_n \, d\Gamma, \quad \forall c_{n+1}^h \in \mathcal{V}^h, \quad (5)$$

where $\Delta t$ is the time increment, $c_n^h$ is the approximate solution at $t_n$, and $\mathcal{V}^h$ is a sufficiently smooth function space. In the standard Galerkin FEM, $\mathcal{V}^h$ is the space of the Lagrangian shape functions used for interpolating $c_{n+1}^h$.

One of the major challenges involved in simulating transient diffusion problems such as the pitting corrosion using the FEM is to maintain a conforming mesh with proper element aspect ratios during the evolution of the
domain morphology. It is evident that the quality of the FE mesh has a crucial impact on the fidelity of results and in particular on the accurate recovery of the gradient field. Note that in diffusion-controlled pitting corrosion problems, the concentration gradient along the pit interface is directly used in (3) to evaluate the moving boundary velocity $v_n$. Thereby, the accurate approximation of the gradient in such problems, which is contingent upon creating elements with small aspect ratios at each time step, has an even more pronounced impact on the accuracy. In the following two sections, we present a computational framework relying on the CISAMR technique for simulating transient moving boundary problems.

3. Automated meshing and mesh evolution

3.1. CISAMR algorithm

Figure 2 schematically illustrates the non-iterative process of transforming a structured mesh (Figure 2a) to a conforming mesh for modeling a corrosion pit using CISAMR. This process involves three major steps: (i) $h$-adaptive refinement of the background mesh using a customized Structured Adaptive Mesh Refinement (SAMR) algorithm, as shown in Figure 2b; (ii) $r$-adaptivity of the nodes of elements intersecting with the materials interface, as shown in Figure 2c; (iii) Sub-triangulating the remaining nonconforming elements, together with elements with hanging nodes on their edges, as shown in Figure 2d. A detailed description of the CISAMR algorithm for modeling multi-phase and fracture problems is presented in [86, 93]. For completeness, before discussing the application of this method for simulating moving boundary problem, here we provide a brief overview of this non-iterative mesh generation algorithm.

- $h$-adaptivity: As the first step, we implement a customized SAMR algorithm to achieve the desired level of refinement in the vicinity of the materials interface (Figure 2b). At each level of refinement, the quadrilateral background elements cut by the interface and their selected neighboring elements are recursively subdivided into four sub-quadrangles. A neighboring element is subjected to refinement if one of its nodes $N_i$ is located at distance $d < 0.5h$ from the intersection point of the interface with one of the edges connected to that ($h$ is the length of that edge). To ensure that the conforming elements of the final conforming mesh have proper aspect ratios, any element with more than one hanging node on one of its edges, which could be created due to multiple levels of SAMR in adjacent elements,
is also subjected to refinement. It is worth mentioning that the objectives of the local refinement of the mesh in the vicinity of the interface during the SAMR phase are twofold: (i) reducing the geometric discretization error and (ii) more accurate approximation of the field and its gradient in this region. Note that the accurate prediction of the gradient field is crucial for the accurate evaluation of the moving boundary velocity in diffusion-controlled pitting corrosion problems. For this purpose, one can either increase the number of SAMR levels near the interface, as used in the current work, or alternatively implement a $p$-adaptive refinement strategy.

- $r$-adaptivity: Next, we visit the nodes of refined elements cut by the
interface to determine their new locations based on their relative distance to the moving boundary interface. It must be emphasized none of the nodes of elements not intersecting with the interface are relocated during the \( r \)-adaptivity, which highly facilitates the implementation of this non-iterative step. As illustrated in Figure 2c, performing the \( r \)-adaptivity leads to deforming the original nonconforming elements of the background mesh, which transforms some of them to conforming elements, while the rest are converted to elements diagonally cut by the interface. The following algorithm, which is schematically shown in Figure 3, is employed to determine the new location of node \( N_i \) of a nonconforming background element during this process:

1. If none of the edges connected to \( N_i \) is cut by the interface: the node is not relocated.

2. If only one of the edges connected to \( N_i \) is cut by the interface: (a) If \( d \geq 0.5h \) the node is not relocated. (b) If \( d < 0.5h \) we snap \( N_i \) to the edge/interface intersection point.

3. If two of the edges connected to \( N_i \) are cut by the interface: (a) If \( d_1 \geq 0.5h_1 \) and \( d_2 \geq 0.5h_2 \) the node is not relocated. (b) If \( d_1 < 0.5h_1 \) and \( d_2 \geq 0.5h_2 \) we snap \( N_i \) to the edge/interface intersection point at \( d_1 \). (c) If \( d_1 < 0.5h_1 \) and \( d_2 < 0.5h_2 \) and \( d_1 < d_2 \) we snap \( N_i \) to the edge/interface intersection point at \( d_1 \) and discard the other intersection point.

It is worth mentioning that none of the case scenarios outlined above contradict or interfere with one another; thus each node can be visited independently during the \( r \)-adaptivity process to determine its new location. Also, note that weak discontinuities (sharp corners) in the materials interface can easily be handled via a hierarchical \( r \)-adaptivity scheme, as shown in Figure 4a. In this non-iterative approach, we first apply the regular \( r \)-adaptivity algorithm described above to nodes of the element holding the weak discontinuity point. We then identify the node that has the closest distance to the gradient discontinuity point, followed by relocating the identified node to this point. It must be noted that in modeling pitting corrosion problems, such weak discontinuities often occur at the intersection point of the pit interface and the metal surface protected by the passive film, as depicted in Figure 1.

- **Sub-triangulation:** The final step of the CISAMR algorithm is to sub-triangulate the background elements that either are deformed after the
Case ii(a) Do not move
Case ii(b) 
Case iii(a) Move to intersection point
Case iii(b) Move to point at distance $d_1$
Case iii(c) Move to point at distance $d_2$
Discard point at distance $d_2$

Figure 3: Different case scenarios for relocating a mesh node during the $r$-adaptivity phase of CISAMR.

(a) Hierarchical $r$-adaptivity of nodes of an element cut by a weakly discontinuous materials interface; (b) single- and double-diagonal rules for sub-triangulating a deformed background element.

$r$-adaptivity or have hanging nodes (generated during the SAMR process) on their edges, as shown in Figure 2d. Sub-triangulating the background elements with hanging nodes is a straightforward task, which depending on the number of hanging nodes would result in the construction of three to six sub-elements. For sub-triangulating the elements with at least one re-located node after applying the $r$-adaptivity, the following simple rules ensure that the aspect ratios of resulting sub-elements are lower than three (Figure 4b):

1. **Single-diagonal rule**: If the element is not cut by the interface along the diagonal emanating from its smallest angle $\theta_{\min}$, the other diagonal (corresponding to the largest angle $\theta_{\max}$) is employed to cut the
elements into two conforming sub-triangles.

2. **Double-diagonal rule**: Otherwise, if $\theta_{\text{min}} < 60^\circ$, the elements is sub-triangulated by cutting that along both diagonals. In this case, an additional interior node is created at the intersection point of the interface with the diagonal emanating from $\theta_{\text{max}}$, which is used for subdividing the background element into four conforming sub-triangles.

### 3.2. CISAMR modeling of moving boundary problems

One of the unique advantages that CISAMR shares with X/GFEM for modeling moving boundary problems is that only the background elements cut by the materials interface must be modified for the reconstruction of the mesh. This not only facilitates the remeshing process but also reduces the computational burden and the error associated with mapping nodal values of the solution between the deformed and updated meshes at each time step. As shown in Figure 5a, the simulation begins with determining new coordinates of the interface nodes at $t_{n+1}$ as

$$
x_{t_{n+1}} = x_{t_n} + (v_n \Delta t) n.
$$

In an activation-controlled pitting corrosion problem, a constant velocity evaluated using (4) is assigned to each interface node. As the ions concentration along the interface reaches the saturation concentration or subject to a sufficiently high overpotential, the corrosion regime becomes diffusion-controlled and the interface velocity must be evaluated using (3), which is a function of the concentration gradient. Because CISAMR ensures that the adaptively refined conforming elements generated along the interface have proper aspect ratios, the gradient and thereby the interface velocity can accurately be approximated without using an advanced smoothing algorithm. Thus, we simply evaluate the gradient at quadrature points of the elements located along the pit interface and then project them to interface nodes. The nodal averaging scheme is then employed to approximate the gradient used for evaluating the interface velocity $v_n$ at each node.

After evaluating the new locations of interface nodes at $t_{n+1}$, one can use a standard time integration scheme (e.g., the backward Euler method) to approximate the field at this time step. However, the continuation of this process in subsequent time steps could result in significant deterioration of the element aspect ratios and even the tangling element issue. This is one of the major challenges that necessitates additional treatments (e.g., node
Figure 5: (a) Deformed mesh created after moving the mesh nodes located along the pit interface in the direction of their outward unit normal vectors to evaluate the solution at $t_{n+1}$; (b) The new mesh reconstructed using CISAMR at $t_{n+1}$, which requires projecting the nodal values at the new nodes (shown in red) generated during this process.

relocation or mesh reconstruction) in methods such as ALE. In addition to the computational cost associated with either of these approaches, one must re-evaluate field values at the nodes of the new mesh based on the solution approximated using the deformed mesh at $t_{n+1}$, which are then used as initial values for evaluating the field at $t_{n+2}$. Since the FE solution is super-convergent at mesh nodes, mapping the solutions between all the nodes of deformed and updated meshes at each time step leads to the loss of accuracy and deterioration of the convergence rate [42].

To simulate a moving boundary problem using CISAMR, we can regenerate the mesh at every time step to avoid the deterioration of elements quality as the solution proceeds. If the maximum relocation distance of an interface node is small compared to the element size, the mesh reconstruction could happen after a few time steps. Unlike conventional mesh generation algorithms, new nodes are only created along the interface in the mesh reconstructed using CISAMR. To generate this mesh, we employ NURBS to parameterize the moving interface geometry as an explicit function. Using a NURBS curve of at least second-order provides a $C^1$-continuous representation of the interface geometry that yields a unique normal vector $\mathbf{n}$ at each interface node, which in turn improves the accuracy for approximating $v_n$.

A NURBS curve $C(u, t)$ representing the moving boundary in this work is composed of $n$ B-splines $M_i^p(u)$ of order $p$ that are functions of parametric
coordinate \( u \) and time \( t \). A set of breakpoints \( U = \{u_1, u_2, u_3, \ldots, u_{n+p+1}\} \) define the region the parametric coordinate is defined on, as well as potential points with gradient discontinuity. A set of control points \( \mathbf{x}_i(t) \) with a given weight \( w_i(t) \), defined in the cartesian coordinate system, are then used to determine the shape of \( \mathbf{C}(u, t) \) as \[ C(u, t) = \sum_{i=1}^{n} \frac{\mathbf{x}_i(t)w_i(t)M_i^p(u)}{\sum_{j=1}^{n} w_j(t)M_j^p(u)}. \] (7)

Note that in the equation above, both \( \mathbf{x}_i(t) \) and \( w_i(t) \) affect the shape of the NURBS curve. Also, these parameters are considered as time-dependent parameters to take into account the evolution of the moving interface geometry. While not investigated in the current manuscript, this NURBS-based explicit representation of the interface can handle topology changes such as interaction between multiple moving boundaries (e.g., growing corrosion pits) by merging interface nodes associated with intersecting NURBS curves.

After updating the NURBS parameterization of the moving boundary at \( t_{n+1} \), we interact this curve with the background mesh to create a new conforming mesh using the CISAMR non-iterative algorithm described previously. It is worth mentioning that the remeshing in CISAMR is a recursive (and not iterative) process that only affects the elements adjacent to the moving boundary. The key advantage of this method appears during the projection of nodal values of the field between the old (deformed) and new (reconstructed) meshes that coexist at \( t_{n+1} \). Figure 5b illustrates the conforming elements in the new mesh constructed by CISAMR at \( t_{n+1} \) for the updated interface morphology shown in Figure 5a. Comparing the two meshes depicted in these figures indicates that nodal values of the field must only be interpolated at certain nodes of the elements located along the interface (shown in red in Figure 5b). All other nodes of the reconstructed mesh away from the interface directly inherit solution values from the nodes with similar coordinates in the deformed mesh depicted in Figure 5a.

Restricting the projection to a minimum number of nodes near the interface, at which this interpolation would be inevitable due to evolution of the geometry, reduces the computational cost and could potentially improve the accuracy as well. Note that for a problem with a non-prescribed interface velocity, which is evaluated as a function of the gradient along the interface (Eq. 3), performing the nodal projection along the interface could take away the second advantage. Within the CISAMR framework, this error can be
minimized by increasing the number of SAMR levels along the interface to restrict the inverse mapping to a thin layer of elements in this region.

In order to project the field values to the nodes of the reconstructed mesh that are located near the boundary (red nodes in Figure 5b), we first locate the element in the deformed mesh that contains each node. Assuming that the global coordinates of node \( N_{\text{new}} \) that requires nodal projection is \((x_p, y_p)\), we identify node \( N_{\text{old}} \) belonging to the deformed mesh that has the closest distance to \( N_{\text{new}} \). We then map \((x_p, y_p)\) to local coordinates \((\xi_p, \eta_p)\) of all the elements sharing node \( N_{\text{old}} \) using an inverse isoparametric mapping which can be written as

\[
(x_p, y_p) = \sum_{i=1}^{n_n} N_i(\xi_p, \eta_p)(\xi_i, \eta_i),
\]

where \( N_i \) is the Lagrangian shape function corresponding to the \( i \)th node of the element, \((\xi_i, \eta_i)\) is the local coordinate of this node, and \( n_n \) is the number of the nodes of this element. After evaluating \((\xi_p, \eta_p)\) from (8), the element in the deformed mesh at which conditions \( 0 \leq \xi_p \leq 1 \) and \( 0 \leq \eta_p \leq 1 \) are satisfied is identified as the element holding \( N_{\text{new}} \). We can then easily interpolate the solution at \((\xi_p, \eta_p)\) based on nodal solution values of this element using an isoparametric mapping.

### 3.3. Implementation aspects

It is worthwhile to further discuss two implementation issues related to the application of CISAMR for simulating moving boundary problems. The first issue pertains to the projection algorithm described above, as \( N_{\text{new}} \) may fall outside the discretized boundary of the deformed mesh. As shown in Figure 6a, this could happen due to the geometric discretization error associated with different discretization patterns of the old and new meshes. Although by using sufficient levels of refinement during the SAMR phase this geometric discretization error becomes negligible, conditions \( 0 \leq \xi_p \leq 1 \) and \( 0 \leq \eta_p \leq 1 \) may not be satisfied in any of the elements sharing \( N_{\text{new}} \) after the inverse mapping. In such cases, we expand the search by introducing a small tolerance \( \varepsilon > 0 \) on upper and lower bounds of these conditions (e.g., \( -\varepsilon \leq \xi_p \leq 1 + \varepsilon \)) to locate the element closest to the projected node. Note that \( N_{\text{new}} \) is slightly outside the edges of the element identified using this approach. However, the isoparametric mapping still can be employed to project the field value at \( N_{\text{new}} \) using the nodal values of this element with a negligible error, provided that the geometric discretization error is also negligible.
Figure 6: (a) Lower right portion of the corrosion pit depicted in Figure 5, showing a new node belonging to the reconstructed mesh that falls outside the discretized domain of the old (deformed) mesh after mapping; (b) upper left portion of the pit in Figure 5a, showing the moving boundary velocity at the intersection point of the pit and the metal surface.

The second implementation issue that worths discussing here is related to calculating the moving boundary velocity at the intersection point of the pit interface and the metal surface covered with a protective film in pitting corrosion problems (Figure 6b). The gradient discontinuity of the domain boundary at this point leads to a higher error in recovering the gradient, which in turn deteriorates the accuracy of the moving interface velocity \( v_n \) evaluated using (3) in diffusion-controlled pitting corrosion problems. What increases the error in computing \( v_n \) at this point even further is the difference between the boundary conditions imposed along the metal surface (non-corroding, \( \nabla c \cdot n = 0 \)) and the pit interface (corroding, \( \bar{c} = c_{\text{sat}} \)). Hence, recovering the gradient at this point via the nodal averaging scheme and using that in (3) to evaluate \( v_n \) leads to an under-estimation of the velocity, which would deteriorate the accuracy as the simulation proceeds.

To resolve this issue, we assume that \( v_n \) at the intersection point equals to \( \bar{v}_n \), which is the normal speed of the closest interface node to the intersection point. This assumption allows for a more accurate approximation of \( v_n \) and is valid when the two points are sufficiently close. Instead of moving the corner points in their normal directions, we move them in the direction of the metal surface (horizontal direction in Figure 6b) to avoid detachment between the moving interface and the non-corroding metal surface. The projection of the horizontal speed of a corner point (\( v_c \)) on the outward normal vector \( n_c \) at this corner point is equal to \( v_n \), which yields the following expression for evaluating \( v_c \)

$$v_c = \frac{\bar{v}_n}{|n_c \cdot e_1|}.$$  \hspace{1cm} (9)
4. Numerical Examples

Five example problems are provided in this section to demonstrate the application of CISAMR for modeling moving boundary problems. In the first three examples we study the convergence rate and verify the accuracy of simulations. The last two examples show the application of CISAMR for simulating corrosion problems with more complex evolving morphologies.

4.1. Convergence study

In this example, we study the accuracy and convergence rate of CISAMR for modeling a moving boundary diffusion problem with the geometry and boundary conditions depicted in Figure 7a. The circular region located at the center of this 5 mm × 5 mm square domain has an initial radius of $R(0) = 960 \, \mu m$ and an initial concentration of $c(x,0) = \exp(-|x|^2/4D)$. Analytical solutions for the evolution of the field and radius of the circle is given by

$$c(x,t) = \frac{1}{t + 1} \exp\left(-\frac{x \cdot x}{4D(t + 1)}\right),$$

$$R(t) = \sqrt{-4D(t + 1) \ln[\bar{c}(t + 1)]},$$

where $D = 0.05 \, \text{mm}^2/\text{s}$ and $\bar{c} = 0.01$ mol/L is the prescribed Dirichlet boundary condition along the perimeter of the circle. To simulate the problem using a prescribed velocity along the circular-shaped moving boundary, (11) is employed to analytically evaluate the normal component of the interface velocity as

$$\bar{v}_n = \frac{\partial R(t)}{\partial t} = -\frac{2D}{R(t)} \{\ln[\bar{c}(t + 1)] + 1\}.$$ 

The CISAMR approximation of the field at $t = 0.6$ s using a 30 × 30 background mesh for discretizing the domain with two levels of SAMR is depicted in Figure 7b ($\Delta t = 0.05$ s). The radius of the circle at this time step has approximately increased by 20%. The other three quarters of Figure 7b characterize the quality of the conforming elements generated using CISAMR based on three metrics: aspect ratio $R_a$, edge ratio $R_e$, and minimum angle $\theta_{\min}$. The first two metrics are evaluated as [86]

$$R_a = \frac{L_{\max} \sum_{i=1}^{3} L_i}{4\sqrt{3}A}, \quad R_e = \frac{L_{\max}}{L_{\min}},$$

17
where $L_i$ and $A$ are the length of the $i$th edge and area of the element, respectively. This figure confirms that the CISAMR non-iterative algorithm can generate high-quality conforming elements with $\max(R_a) = 1.84$, $\max(R_e) = 2.41$, and $\min(\theta_{\text{min}}) = 25.8^\circ$.

![Figure 7](image)

**Figure 7:** First example problem: (a) schematic of the domain geometry and initial/boundary conditions; (b) CISAMR approximation of the field (top left quarter) and three mesh quality metrics at $t = 0.6$ s using a $30 \times 30$ background mesh with two levels of SAMR for discretizing the domain ($\Delta t = 0.05$ s).

Variations of the $L_2$ and $H^1$ norms of the error versus the element size $h$ for the CISAMR approximation of this problem using a prescribed interface velocity is presented in Figure 8a. At a given time step, these norms of the error are computed as

$$E_{L_2} = \sqrt{\int_{\Omega} (c_{\text{ex}} - c^h)^2 \, d\Omega}, \quad E_{H^1} = \sqrt{\int_{\Omega} \left[ (c_{\text{ex}} - c^h)^2 + (\nabla c_{\text{ex}} - \nabla c^h)^2 \right] \, d\Omega},$$

where $c_{\text{ex}}$ and $c^h$ are the exact and approximate fields, respectively. Five structured meshes with 10, 20, 30, 40, and 50 elements along each edge are employed to create conforming meshes with 0, 1, and 2 levels of SAMR. In the absence of $h$-adaptive refinement, CISAMR yields optimal convergence rates, i.e., 2.01 and 1.03 for the $L_2$ and $H^1$ norms of the error, respectively (measured between last two data points). Due to the creation of smaller conforming elements along the interface for meshes with one and two levels of
SAMR, the improved convergence rates in the corresponding plots in Figure 8a is not meaningful. However, it can be seen that the adaptive refinement along the interface has a notable impact on reducing the $L_2$-norm of the error. For example, the $L_2$-norm of the error associated with the $50 \times 50$ background mesh with two levels of SAMR is approximately 174% lower than that corresponding to the same mesh with no adaptive refinement.

![Figure 8: First example problem: variations of the $L_2$ and $H^1$ norms of the error versus the element size $h$ for three different levels of SAMR at $t = 0.6$ s in simulations with (a) prescribed interface velocity; (b) non-prescribed interface velocity evaluated using (15).](image)

As shown in Figure 8a, except for the models built using the coarsest mesh ($10 \times 10$), increasing the number of SAMR levels has no appreciable impact on reducing $E_{H^1}$ in simulations with prescribed interface velocity. Note that according to (10), the maximum curvature of $c(x, t)$ occurs at the center of the circular-shaped domain. Thus, the highest values of the error associated with the gradient recovery is observed in the elements located near this region, which are away from the refinement zone. Hence, when the $H^1$-norm of the error is integrated over the entire domain, the dominating errors in these coarse elements hides the improvement achieved near the interface. Since the recovered gradients in elements adjacent to the interface are directly used to evaluate the moving interface velocity in a non-prescribed velocity case scenario, it is worthwhile to study the effect of SAMR on this localized error. Figure 9 shows the distribution of the element-wise $H^1$-norm of the error normalized by the area of each element ($E_{H^1}^h$) at $t = 0.6$ s for
simulations with different levels of SAMR applied to a 20 × 20 background mesh. Given the high values of $E_{H^1}^b$ in the rectangular elements near the central region, they are excluded from the figures to better visualize the magnitude of error in the remaining elements. Figure 9 clearly shows the effectiveness of increasing the number of SAMR levels in reducing the error within the refined regions. For example, the maximum value of $E_{H^1}^b$ in the elements located along the moving boundary of the mesh with two levels of SAMR is approximately 7 times lower than that of the unrefined mesh.

![Figure 9: First example problem: Distribution of element-wise $H^1$-norm of the error normalized by the area of each element at $t = 0.6$ s for simulations with prescribed interface velocity. The CISAMR models are generated using a 20 × 20 background mesh with (a) zero; (b) one; and (c) two levels of SAMR.](image)

As mentioned above, the impact of the SAMR phase on improving the accuracy of CISAMR simulations is more pronounced when non-prescribed boundary velocity must be evaluated as a function of recovered gradients in the elements adjacent to the moving boundary. To better show this effect, we performed a second set of simulations for the current example problems using a non-prescribed velocity $v_n$ approximated as

$$v_n = \frac{\bar{c}}{(t + 1) \nabla c \cdot n} \{\ln[\bar{c}(t + 1)] + 1\}, \quad \forall x \in \Gamma_s,$$

(15)

in which the relation $\nabla c \cdot n = -\frac{R(t)e}{2D(t+1)}$ along the interface is used to replace $R(t)$ in (12). Variations of the $L_2$ and $H^1$ norms of the error versus $h$ for the same background meshes and SAMR levels as those of the prescribed velocity cases corresponding to these simulations are depicted in Figure 8b. As expected, using an approximate normal velocity to evolve the domain

20
geometry at each time increment leads to a sub-optimal convergence rate of 1.72 for the $L_2$-norm of the error. This is because the velocity evaluated using (15) is a function of the recovered gradient in the elements located along the interface, which has a slower convergence rate than the approximate field. Thus, the accumulation of the error corresponding to predicting the evolving domain morphology throughout the simulation disturbs the optimal rate of convergence in the simulations with a non-prescribed interface. Due to the local nature of this error, SAMR of the domain in the vicinity of the interface has a considerably more pronounced impact on reducing $E_{L_2}$ compared to the prescribed velocity cases. For example, as shown in Figure 8b, applying two levels of SAMR to the $50 \times 50$ background mesh leads to more than 243% decrease in $E_{L_2}$ compared to the mesh with no refinement. As shown next, this significant improvement in $E_{L_2}$ is mainly due to a more accurate approximation of the domain morphology as a result of the reduced error in evaluating $v_n$. It is worth mentioning that we can still recover the optimal convergence rate for $E_{H^1}$, although values of the error are larger than those of the simulations with a prescribed interface velocity.

To shed more light on the effect of SAMR on reducing gradient-based local errors, Figure 10a illustrates the variation of the relative $l_2$-norm of the error in approximating the radius versus the element size $h$ (non-prescribed velocity). This norm of the error can be computed as

$$E_R = \sqrt{\frac{\sum_{i=1}^{n_{in}} (R_{\text{ex}} - R_i^h)^2}{(n_{in}R_{\text{ex}})^2}},\tag{16}$$

where $n_{in}$ is the number of nodes located along the interface at $t = 0.6$ s, $R_i^h$ is the approximated radius at node $i$, and $R_{\text{ex}}$ is the exact radius. As shown in Figure 10a, increasing the number of SAMR levels has a pronounced impact on obtaining a more accurate approximation of the domain geometry by reducing the error associated with evaluating $v_n$. For example, the FE simulation conducted on the $30 \times 30$ background mesh with two levels of SAMR yields a 542% lower $E_R$ compared to the case with no SAMR. Effect of the adaptive refinement on approximating the domain geometry at $t = 0.6$ s using this background mesh is better illustrated in Figure 10b. This figure also shows a significant reduction in $E_{H^1}^h$ associated with the elements adjacent to the boundary when increasing the number of SAMR levels. The reduced error in the gradient recovery leads to a more accurate approximation of the moving interface velocity, which in turn yields a more accurate
approximation of the domain morphology for higher levels of SAMR.

Figure 10: First example problem: (a) variation of the relative $l_2$-norm of the error $E_R$ associated with approximating the domain radius versus the element size $h$ for three different levels of SAMR at $t = 0.6$ s; (b) domain geometry and distribution of the area normalized $H^1$-norm of the error in the elements adjacent to the moving boundary for the simulations conducted on a $30 \times 30$ background mesh with different levels of SAMR.

4.2. Semi-circular pit growth

In this example, we employ CISAMR to simulate the diffusion-controlled and activation-controlled growth of a semi-circular shaped corrosion pit in a stainless steel plate, as shown in Figure 11a. For the diffusion-controlled case, $\bar{c} = 0$ mol/L and $\bar{c} = c_{\text{sat}}$ are assigned along the electrolyte and the corroding pit boundaries, respectively. Also, a zero flux boundary condition ($\vec{r} \cdot \vec{n} = 0$) is imposed along the horizontal metal surface to replicate a protective passive film on this surface. We assume that the initial ions concentration in the electrolyte solution is zero everywhere, i.e., $c(x, 0) = 0$ mol/L. For the activation-controlled problem, we replace the Dirichlet boundary condition along the pit interface with the Robin boundary condition given in (2), assuming a uniform overpotential of $\eta = 63$ mV. This overpotential corresponds to a normal component of the interface velocity of $v_n \approx 2$ nm/s, which is evaluated using (4).

The main objective of the current example is to verify the accuracy of CISAMR simulations by comparing the results with similar simulations conducted using the phase field corrosion model presented in [83]. The phase field
Figure 11: Second example problem: (a) schematic of the domain geometry; (b) comparison between the predicted pit depth growth versus time during the diffusion-controlled pitting corrosion using CISAMR and phase field techniques.

method simulates the growth of a corrosion pit using an auxiliary phase field variable to track the evolution of the moving pit interface. In the context of FEM, using the phase field variable enables the use of a nonconforming mesh for discretizing the domain, although it increases the computational cost due to the requirement to solve for an additional variable at each time step. To obtain a reference solution using the phase field model to compare with CISAMR results, we use a refined mesh composed of six-node Lagrangian elements with a maximum size of 25 \( \mu \text{m} \) to discretize the domain. Adaptive mesh refinement is employed in the vicinity of the pit interface to more accurately approximate sharp gradients in this region, resulting in 350,534 degrees of freedom (DOFs) in the first time step. Note that as the pit grows, the number of DOFs constantly increases in subsequent time steps. The phase field simulations are conducted using an adaptive time marching scheme with a maximum allowable time step of \( \Delta t = 0.5 \text{ s} \). It is worth mentioning that the first few time increments needed for convergence in these simulations are considerably smaller (\( \Delta t \approx 10^{-7} \text{ s} \)). Also, in order to accurately approximate the sharp gradients of the field within the diffuse interface, the mesh is highly refined in this region to create at least six nodes across its thickness.

A comparison between CISAMR and phase field approximation of the pit depth evolution versus time in the diffusion-controlled problem is presented in
Figure 11b. The CISAMR simulation is conducted using a $30 \times 30$ background mesh for discretizing the domain and a time increment of $\Delta t = 2.5$ s. Also, two levels of adaptive refinement is applied to the mesh in the SAMR phase. Note that the element size near the interface after this adaptive refinement is still approximately 5 times larger than the (quadratic) element sizes within the diffuse interface in the phase field model. A comparison between the FE meshes for discretizing the rectangular region in Figure 11a is provided in Figure 12a. The material properties used in this example and all the following examples (unless stated otherwise) are given in Table 1.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_{\text{solid}}$</td>
<td>143 mol/L</td>
<td>$c_{\text{sat}}$</td>
<td>5.1 mol/L</td>
</tr>
<tr>
<td>$D$</td>
<td>$8.5 \times 10^{-10}$ m$^2$/s</td>
<td>$n$</td>
<td>2.1 equiv/mol</td>
</tr>
<tr>
<td>$F$</td>
<td>96,485 C/equiv</td>
<td>$i_0$</td>
<td>1.422 mA/cm$^2$</td>
</tr>
<tr>
<td>$b_0$</td>
<td>45.1 mV</td>
<td>$\eta$</td>
<td>63 mV</td>
</tr>
</tbody>
</table>

Table 1: Values of the parameters used in the pitting corrosion example problems.

As shown in Figure 11b, both CISAMR and the phase field methods yield similar approximations of the pit depth at any given time, with a maximum relative error of 2.3%, which happens at $t \approx 60$ s. Note that while the CISAMR simulation starts with a prescribed concentration of $\bar{c} = c_{\text{sat}}$ along the pit interface, the phase field method does not allow prescribing this Dirichlet boundary condition. Therefore, the concentration along the pit interface in the phase field model ramps up from 0 mol/L to $c_{\text{sat}}$ in the first few time steps, which accounts for the initial discrepancy between the results shown in Figure 11b. However, due to the diffusion of ions away from the pit bottom, this initial effect fades in subsequent time steps, resulting in a considerably smaller relative error of only 0.19% at the last time step ($t = 400$ s). A comparison between morphologies of the pits obtained from the CISAMR and phase field simulations at this time step is provided in Figure 12b.

Figure 12c illustrates the CISAMR approximations of the ions concentration and the pit morphology at the last time step ($t = 3.89$ h) of the activation-controlled pitting problem. This figure also shows the corresponding phase field simulation, which predicts a similar distribution of the ions concentration inside the pit. Note that the maximum ions concentration in this case ($c_{\text{max}} = 0.078$ mol/L) is well below $c_{\text{sat}}$ and thereby the activation-
Figure 12: Second example problem: (a) FE meshes corresponding to the rectangular region shown in Figure 11a for the CISAMR and phase field simulations; comparison between the resulting evolved pit morphologies at the last time step for (b) diffusion-controlled pitting, $t = 400$ s; (c) activation-controlled pitting, $t = 3.89$ h.
controlled assumption in these simulations is valid. Further, the pit maintains its semi-circular shape, as a constant velocity obtained from (4) is assigned along the pit interface throughout the simulation. This velocity is considerably smaller than the diffusion-controlled velocity given by (3) and thereby a larger time step \((\Delta t = 100 \text{ s})\) is used for the CISAMR simulation of this activation-controlled problem. Note that throughout both the diffusion-controlled and activation-controlled simulations, the maximum aspect ratio of conforming elements generated using CISAMR is lower than 1.70.

4.3. Modified pencil electrode test

In this example, we use CISAMR to simulate a modified pencil electrode corrosion test with the initial and boundary conditions shown in Figure 13a. In the pencil electrode test, the metal wire and the electrolyte are surrounded by epoxy layers, which simulate an insulated boundary condition for the concentration field. Under the diffusion-controlled condition, the pit surface remains flat and only moves in the longitudinal direction. A standard pencil electrode test is designed such that the pit mouth has the same concentration as the bulk solution \([83, 95]\). In this example, we assume that the pit mouth is covered (e.g., with a porous glass separator) such that the exchange of water between the bulk and pit solution is allowed but ions cannot be transferred outside the pit. Therefore, the growing pit is always filled with the electrolyte and the average ions concentration is constantly increasing until it reaches \(c_{\text{sat}}\). Since according to (3) the mass transfer is the rate-limiting step in diffusion-controlled pitting corrosion, after reaching \(c_{\text{sat}}\) everywhere in the pit, \(\nabla c(x, t) \cdot n = 0\) along the pit interface and therefore the pit can no longer grow.

Using the mass conservation law, we can analytically determine the final depth \(d_f\) of the pit as

\[d_f = \frac{c_{\text{solid}} d_i}{c_{\text{solid}} - c_{\text{sat}}},\]

where \(d_i\) is the initial pit depth. In this example, the metal wire has a radius of 100 \(\mu\text{m}\) and the initial pit depth is \(d_i = 301 \mu\text{m}\). To perform the simulation, an \(80 \times 16\) background mesh with two levels of refinement is employed to build the FE mesh using CISAMR (time increment: \(\Delta t = 4 \text{ s}\)). We assume \(c_{\text{solid}} = 30 \text{ mol/L}\) to allow an appreciable growth of the pit depth before being fully saturated. The simulated variation of the pit depth versus time is depicted in Figure 13b, which shows that CISAMR predicts a similar value as the analytical solution evaluated using (17) for
the final pit depth, with a relative error of 0.24%. This figure also shows that the growth rate of the pit is higher at the beginning of the simulation due to a higher concentration gradient along the pit interface. However, the growth rate gradually decreases and eventually stops when the electrolyte is fully saturated with dissolved ions. Three snapshots of the pit geometry in this simulation, together with the evolution of the ions concentration, are illustrated in Figure 13a. Note that at the last time step \( t = 300 \text{s} \), the ions concentration is equal to \( c_{\text{sat}} = 5 \text{ mol/L everywhere inside the pit.} \)

4.4. Electropolishing of stainless steel

In this example, we simulate the electropolishing process in a stainless steel specimen with the geometric features shown in Figure 14. The electropolishing is commonly used in industry to reduce the surface roughness of metallic parts by applying a high electric voltage that induces diffusion-controlled corrosion on the metal surface. The boundary conditions used for simulating this phenomenon is similar to the first example problem with the difference that the zero-flux boundary condition associated with the portion of the metal surface covered by the passive film is eliminated. The CISAMR simulation of the evolving metal surface during the electropolishing process is depicted in Figure 14. As shown in this figure, this diffusion-controlled corrosion phenomenon smoothes the metal surface by eliminating the initial roughness. This is because of the smaller diffusion distance and hence
the higher ions concentration gradient in the vicinity of higher regions of the surface, resulting in a higher corrosion rate near such regions. Figure 14 also illustrates half of the 20 × 20 background mesh used for discretizing the domain, together with the conforming elements generated using CISAMR at each snapshot. The maximum element aspect ratio recorded during this simulation is 1.69.

4.5. Growth of arbitrary-shaped corrosion pits

In this final example problem, we employ CISAMR to simulate the growth of three pits with complex initial morphologies, as shown in Figure 15. The simulations are conducted under both the diffusion-controlled and activation-controlled conditions. The initial and boundary conditions are similar to those described in the second example problem (Figure 1). A 20 × 20 background mesh is employed to create the FE models for all three pits using CISAMR with two levels of SAMR.
Figure 15: Fifth example problem: Domain geometry and initial shapes of corrosion pits.

Figure 16: Fifth example problem: Diffusion-controlled CISAMR simulation of evolving shapes of three corrosion pits with different initial geometries and the corresponding approximated concentration fields.
The CISAMR simulation of the evolving pits morphologies under the diffusion-controlled condition is depicted in Figure 16. As shown in that figure, the diffusion-controlled process leads to opening of the pit mouth, while causing a considerably less growth at the pit bottom. This is due to the lower concentration of ions near the pit mouth compared to the pit bottom, which in turn leads to a higher concentration gradient and thereby higher interface velocity according to (3) in the former region. Note that the low concentration of ions in the bulk electrolyte near the pit mouth accelerates the diffusion of ions away from the interface, which justifies the lower ions concentration observed there. Figure 16 also shows the evolution of the conforming meshes generated using CISAMR for each simulation.

Figure 17 illustrates the CISAMR approximation of the concentration field during the activation-controlled growth of the pits shown in Figure 15. Due to the lower velocity of the pit interface in this condition \( v_n \approx 2 \text{ nm/s} \), these simulations are conducted over a longer time period \( t_{\text{max}} \approx 10 \text{ h} \) to allow an appreciable pit growth. Note that in these simulations, the ions concentration in the electrolyte is built up at the pit bottom, with the maximum value at the deepest point of the pit. This trend is attributed to the slow growth of the pit, which allows the dissolved ions near the pit mouth to fully diffuse into the bulk electrolyte. However, as the pit grows, the increase of the diffusion distance near the pit bottom leads to the increase of the ions concentration in this region, as observed in Figure 17. The maximum element aspect ratio during the evolutions of all six meshes used in the diffusion and activation controlled simulation does not exceed 2.04.

5. Conclusion

The Conforming to Interface Structured Adaptive Mesh Refinement (CISAMR) technique was implemented to simulate moving boundary transient diffusion phenomena, with the activation-diffusion-controlled pitting corrosion selected as the main application problem. CISAMR is a non-iterative algorithm that transforms a structured grid into a high quality conforming mesh using a combination of customized \( h \)-adaptivity, \( r \)-adaptivity, and sub-triangulation algorithms. A key feature that facilitates the application of CISAMR and increases its accuracy for modeling moving boundary problems is that the background mesh is only modified in the vicinity of the moving interface and all of the mesh nodes away from that directly inherit the nodal values of the solution from the undeformed mesh. We discussed several implementation is-
sues pertaining to the application of CISAMR for modeling moving boundary problems and in particular the pitting corrosion phenomenon. In addition to a detailed convergence study, two benchmark problems were presented to verify the accuracy of this method by comparing the results with analytical solutions and phase field simulations. Two example problems were also presented to show the application of CISAMR for simulating an electropolishing process, as well as the activation- and diffusion-controlled growth of arbitrary-shaped corrosion pits.
Acknowledgement

This work has been supported by the Air Force Office of Scientific Research (AFOSR) under grant number FA9550-17-1-0350. The authors also acknowledge partial funding by the Ohio State University Simulation Innovation and Modeling Center (SIMCenter) through support from Honda R&D Americas, Inc.

References


