Modelling and Simulation in Materials Science and Engineering

PAPER • OPEN ACCESS

Interoperability architecture for bridging computational tools: application to steel corrosion in concrete

To cite this article: Zahid M Mir *et al* 2020 *Modelling Simul. Mater. Sci. Eng.* **28** 025003

OPEN ACCESS
IOP Publishing
Modelling and Simulation in Materials Science and Engineering

Modelling Simul. Mater. Sci. Eng. **28** (2020) 025003 (19pp)
https://doi.org/10.1088/1361-651X/ab6209

Interoperability architecture for bridging computational tools: application to steel corrosion in concrete

Zahid M Mir[1,5], Jesper Friis[2], Thomas F Hagelien[3], Ingeborg-Helene Svenum[2], Inga G Ringdalen[2], Natalia Konchakova[1], Mikhail L Zheludkevich[1] and Daniel Höche[1,4]

[1] Institute of Materials Research, Helmholtz-Zentrum Geesthacht, Max-Planck Str. 1, D-21502 Geesthacht, Germany
[2] SINTEF Industry, NO-7465 Trondheim, Norway
[3] SINTEF Ocean, NO-7465 Trondheim, Norway
[4] Computational Material Design, Helmut-Schmidt-University, D-22043 Hamburg, Germany

E-mail: zahid.mir@hzg.de, jesper.friis@sintef.no, thomas.f.hagelien@sintef.no, ingeborg-helene.svenum@sintef.no, inga.ringdalen@sintef.no, natalia.konchakova@hzg.de, mikhail.zheludkevich@hzg.de, daniel.hoeche@hzg.de and hoeched@hsu-hh.de

Received 12 June 2019, revised 21 November 2019
Accepted for publication 16 December 2019
Published 15 January 2020

Abstract

A multiscale modelling framework, especially for corrosion modelling, requires not only robust computational tools but also an efficient datacentric architecture for handling information exchange at different modelling scales. Different computational solvers require and produce data in different programming languages and specific formats signifying a strong non-uniformity for an easy nexus with other solvers. This non-uniformity has created a need to focus on intermittent state-of-the-art datacentric software tools which aim to bridge data exchange heterogeneity across diverse set of solvers. Data organization in the form of metadata structures are presented as a standard for a coherent information representation regardless of the diverse nature of data formats specific to a scientific discipline. This fundamental work presents the

[5] Author to whom any correspondence should be addressed.

0965-0393/20/025003+19$33.00 © 2020 The Author(s). Published by IOP Publishing Ltd Printed in the UK

concept, underlying terminology and working mechanism of a datacentric architecture tool SOFT5 for exchanging and interfacing data-flow between solvers and its present application to a concrete technology multiscale simulation network as a potential application.

Keywords: multiscale framework, data science, datacentric platform, data inter-exchange, metadata structures, semantic interoperability, JSON

(Some figures may appear in colour only in the online journal)

1. Introduction

The recent decades in the modelling science have been marked with an unprecedented growth in the computational power. This fact has been further complemented with a sharp shift towards multiscale approaches regarding a particular domain problem. As such, multiscale scientific programs have gained pace in the recent years across all branches of science. A multiscale approach involves understanding of a particular problem usually stretching across multiple modelling scales ranging from electronic-atomic level to macro-structure level. Such simulation strategies involve use of multiple software and solvers with the general aim of approaching the same problem at different levels of understanding (nanoscale to macroscale).

Each modelling scale in time and space is usually associated with at least one computational solver often forming an overall pipeline structure across all scales. At each scale, data is extracted and exchanged. Thus, these solvers are associated with handling and processing of enormous amounts of data from different sources and in varying formats. As an example, the process simulator in a chemical manufacturing plant would typically involve exchange of data ranging across varying time and spatial domains. An extension of such a flow simulator structure by incorporating additional sub-modules and facilitating coherent data exchange in the modified structure via a generic datacentric platform would be a very efficient approach. However, least attention is usually paid in terms of data science, which is essential in bridging computational tools employed at different levels of modelling.

Classifying data is a basic strategy for data exchange. Usually data originates from various different sources such as open or propriety based software and in various formats from closed-custom formats to commonly used open standard formats. Moreover, data is usually incomplete in its understanding such as it lacks supporting data ‘metadata’ such as units, data type, description of data and source, making it difficult for other computational models to interpret the data. Therefore, classification of data on a semantic level forms the starting point of data interoperability.

The conceptual use of metadata has been traced back to around 280 BC at the Great Library of Alexandria wherein library contents were tagged to provide information about the author, title and the subject to which the literature belonged [1]. This served in sorting and cataloguing the library contents and helped in directing the users to the books they wanted to read instead of opening and going through all the books. The benefits of using metadata later helped in creating more detailed system of cataloguing in the libraries, such as the dewy decimal system, which has been recently replaced by modern digital database cataloguing systems. Thus, metadata has been used in many different fields i.e. from organizing libraries to listing contents on a compact disc. The purpose of such representation is to provide information about a system, without itself being a part of it, but complementing it to complete its meaning or function. As an example, Standard Book Numbering was created in 1966, which was a 9-digit entity to identify a book [2]. It would give information about a book without itself being a part of the book content. This was later changed to 13-digit entity and is referred to as International Standard Book Number (ISBN). ISBN is a metadata attached to book and contains all the necessary information about the book such as registration country, publisher details as well as the title.

In the last few decades, the continuous evolution of numerical technology has resulted in a huge number of versatile stand-alone computational tools. As such, bridging these numerical tools has become cumbersome and therefore more attention is required to connect these systems using machine interpretable syntactic tools. Data sharing between computational solvers should resort to some agreeable formats and standard content sharing definitions. In this regard, well-established standards such as ISO/IEC 11179 covers all aspects of data sharing in information technology. The standard focuses on entire spectrum of data science such as data classification, relationships, formulations, nomenclature and identification principles and many other aspects of metadata based data science.

Although the field of information technology has since long time resorted to standardization mostly in terms of programming language aspects, it has experienced a significant shift towards use of metadata standards. This had made it quite easier to associate correct description and uniform treatment of data during interoperability operations.

Non-standardization usage of metadata can lead to severe failures as there is a big chance of incorrect or misinterpretation of data associated to metadata. One of the most famous examples is the failure of NASA Mars Climate Orbiter in the year 1999 [3]. It occurred due to misinterpretation of data between two technical teams, one of which assumed data values in SI units and the other in metric units. The result was a disastrous accident and a complete loss of the orbiter. It is important to emphasize that, although the exact and correct value was transferred across the two teams, the interpretation of the quantity itself was different i.e. metadata was not part of the data transfer. Manual interpretation of data should be avoided as far as possible and machine-readable metadata should always accompany data for smooth and secure interoperable operations.

This paper presents an approach towards development of a data handling framework for multiscale simulation programs using SOFT5 tool and develop a strategy of coupling heterogeneous syntactic model data using concept of metadata structures. The entire framework is based on data abstraction at multiple levels as illustrated in figure 1. At the concrete level we have the actual data that is formalized into a representation which can be completely described by the metadata. The metadata itself is described by the metadata schema, which in turn is described by the *basic metadata schema*. The basic metadata schema is capable to describe itself, breaking the sequence of higher and higher levels of abstraction. Additional levels of abstraction are possible, but seldom needed in practice. This hierarchy of abstractions opens possibilities for cross-domain interoperability both between domains using different metadata systems and between different physical domains using SOFT5 for metadata representation. Figure 1 shows the latter case. An example of this is the entity representation of diffusivity at the atomic scale that maps to a different entity describing diffusivity at the microscale level. The basic metadata schema can describe the metadata schemata for each domain that describe the domain-specific metadata. To achieve interoperability between two domains, one still has to implement concrete mappings, but the mappings can be implemented at different levels of abstraction. That both domains has a common root, the basic metadata schema, ensures that it is possible to do the mappings.

This makes it easy and convenient to accomplish data handling and data transfer without losing its meaningful value. Relationships between data can then be formed and a structured multiscale workflow can be established where different modelling software and solvers can be

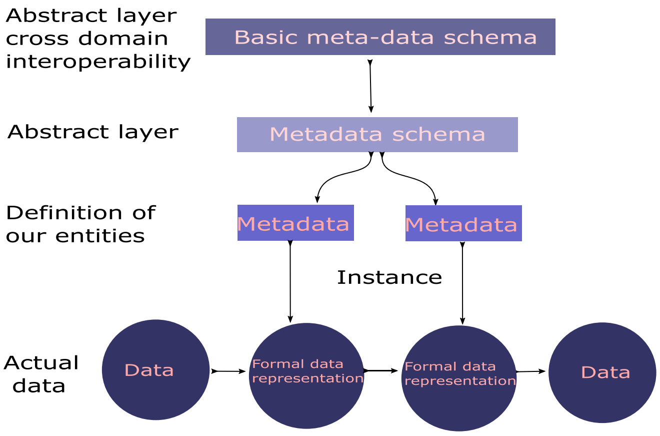

Abstract layer cross domain interoperability
Basic meta-data schema
Abstract layer
Metadata schema
Definition of our entities
Metadata
Metadata
Instance
Actual data
Data
Formal data representation
Formal data representation
Data

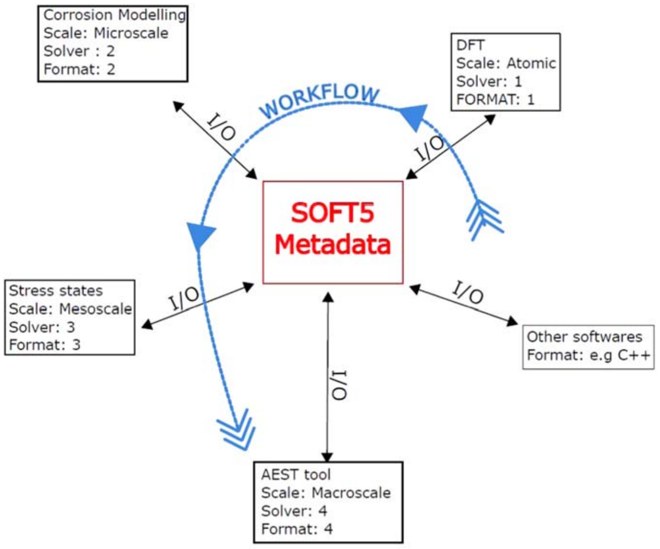
Figure 1. Cross-domain interoperability by metadata, metadata schemata and the basic metadata schema.

connected. Such a framework helps in an easy recognition of domain specific simulation data as well as helps in maintaining trace-ability and version control over it. This results in easier categorization of different domain specific data and facilities fast development of the entire simulation framework.

This paper refers to the modelling framework and methodology adopted in the LORCENIS project [4]. The LORCENIS project focuses on developing specialized concrete recipes with the aim of extending life and imparting additional functionality benefits to energy infrastructure exposed to severe operating conditions. The scales of investigation vary from electronic level and up to macro level. SOFT5 tools has been partially applied for bridging the heterogeneity of different modelling software to facilitate a robust and easy to setup data-centric architecture to handle data flow across the scales. Figure 2 shows the state-of-the-art multiscale materials modelling (M3) framework, that is under development in the LORCENIS project [4], for accurate prediction of service life of structures exposed to corrosive marine environment. This paper presents the direct application of metadata structure and interoperability technology used in the M3 LORCENIS framework.

The modelling architecture is divided into four distinct modelling scales (one discrete and three continuum) ranging from electronic/atomistic level, microscale, mesoscale and eventually to a macroscale level with engineering purpose at the top of modelling chain. A distinct modelling task is assigned to each modelling scale. Furthermore, as illustrated in figure 2, each modelling scale is employing a different computational solver to achieve its modelling goals.

To achieve this task, an efficient data-handling scheme is required to handle the coupling arrangement of these different solvers (open source and propriety based) in order to achieve a smooth compatibility between them. This broad multiscale simulation strategy paves a necessity for a robust state-of-the-art modelling infrastructure. As such, SOFT5 tool has been adopted inside this modelling framework to serve as an interface for data handling between scales. Data sharing aspects are highlighted in this paper, as well as scale based metadata structures, and the working principle are presented. The paper highlights the actual working sequence of SOFT5 inside the various modelling scales along with corresponding programmable metadata structures. The coupling features presented in this study involve data



A graphical explanation of Incorporation and working action of SOFT5 into LORCENIS.

Figure 2. Overall structure of the M3 LORCENIS framework. Picture credits [4].

linking aspects of electronic (discrete) model with microscale (continuum) model as part of framework testing and presenting a proof of concept.

Additionally, a significant point of multiscale modelling concepts is the interoperability between the software used for the computation at difference scales. Usually, different scientific or industrial communities participate in the development of software in order to solve complex materials modelling problems. Thereby they have established different terminologies. The developed modelling codes focus typically on specific application domains and on particular types of models. A general issue is evolving, wherein the different software codes owning specific terminologies of software owners, academic model developers and end-users (SMEs or big companies) for the definition of the same computational problems, but using a different vocabulary. Hence, there is a strong need for standardization. One effort in this direction has been taken by the European Materials Modelling Council (EMMC), who has published CEN Workshop Agreement on modelling terminology, classification and metadata for materials modelling [5]. The document is publicly available as a reference document from the National Members of CEN.

A standardized description of multiscale scientific problems using a uniform vocabulary referred to as materials modelling data table (MODA) [6] is used as key approach to develop and establish successful software interoperability. MODA is an effective instrument for documentation of simulations and understanding of modelling approaches, general project structure as well as the model interaction within workflows. It has established now as a growing industrial and academic standard for computational problems. It is a required tool for the preparation of European research and innovation projects, joint research activity for different modelling scales and application areas.

The use of ontology based organization of scientific data has emerged to be quickly adaptable by many scientific branches such as biomedicine [7] molecular material design and selection [8]. Ontology based data assures interoperability to be easily accessible by users as such mappings and extensions are easily to create on existing data sets. As an example, existing ontology based frameworks include CAPE-OPEN standard [9] which sets standards for computer aided process engineering applications to interoperate. Similarly, Gene Ontology Consortium [10] maintains a comprehensive organization of huge amount of molecular biology experiments.

Ashino [11] has provided a very basic detail of ontology based representation of data. Similarly other works focusing on standardization of data and associated ontology include this [12]. Furthermore a comprehensive take on scientific breakthrough via studying massive data-sets has been provided by Hey *et al* [13]. A basic understanding of technical data is provided by Peter Murray Rust [14]. Cheung [15], Hunt *et al* [16] have identified data science field to be an important asset for the scientific community.

2. Interoperability, metadata and workflow technology

2.1. Interoperability

Interoperability, coupling and linking are latest terminologies in the computational materials modelling community. It is a consequence of the rapid development of numerical software during the last decade with a majority arising from the academic community. The rapidly growing community and the capabilities of computational approaches have thus lead to a wide spectra of models and methods. Software belonging to different domains has been developed independent of each other. Making them communicate is very challenging, since they are based on different concepts and uses different vocabularies. Thus, dedicated tools are required which are feasible to handle different data and data formats. Information exchange which allows software to share data during any kind of simulation step in a efficient way, also requires mature semantic interoperability frameworks and tools [17]. The paradigm change relates to the fact that not information on file formats or protocols are exchanged, but the software systems itself shares relevant information by using a standard semantic framework as a mediator. This framework belongs to an overarching ontology capable to handle all relevant data schemes and metadata respectively. This field of data science is relatively new and often undiscovered respectively and its impact underestimated. Currently, no shared standards for such frameworks exist, but research and funding schemes addressing this issue have been initiated.

2.2. Metadata definitions and concepts

According to the latest guidelines RoMM 6 [6] and [5] from the EMMC[6], metadata is defined as ‘metadata are data that describe and give information about other data’. In other words it is a vocabulary tool for expressing data in a generic way and attaching more information to it for its easier handling [18, 19]. For a particular multiscale modelling scenario, data that is to be exchanged needs to be described completely. This forms the basis of interoperability and data exchange. Metadata structures help in formal description of exchangeable data in a generic sense.

To cite a good example, the field of image-based analysis relies heavily on metadata exchange when images are transferred from one software to another. Here, the EXIF-info attached to the digital images contains information about the image such as name, pixel size, GPS coordinates, image resolution, creation time, date and source. This additional information about the image is nothing but metadata information about the image. Without this

[6] See https://emmc.info/.

metadata, another user who is not the default user of the image would not be able to perform image based analysis as there will be no information on the pixel size and resolution of the image. Hence, quantification of features on the image will be practically impossible to achieve and as such interoperability would be lost. Another example of commonly used metadata is the Digital Object Identifier which acts as a digital identity tag for any digital entity.

With these enhanced data conditioning properties, metadata structures find their broad application across all field of science.

2.3. MODA format for computational problem description

MODA refers to a standardized and concise representation of a simulation problem and the underlying process. With MODA as a standardized terminology, an easier and efficient exchange of a computational methodology could be achieved among users of material modelling codes such as academicians, industrial end-users and experimentalists. MODA therefore facilitates understanding of models by any interested user. MODA tables report on the structure of a simulation and on the relationship between the different model in a workflow.

MODA has the following structure:

- Heading, including name of the user case, project, owner.
- Overview of the simulation, including the chain of models used.
- Workflow, i.e. a graphical representation of the simulation.
- Description of each part of the simulation pertaining to each model used in the chain.

Each MODA should report on these mentioned aspects. A graphical representation of the workflow is also accompanied with MODA and helps in a faster understanding of the workflow path. For a deeper understanding of MODA, the reader is referred to [6]. An user-case example employing MODA is presented later.

2.4. SOFT

SOFT is an acronym for SINTEF Open Framework and Tools. SOFT5 is a set of open source libraries and tools to support scientific software development. The development of SOFT5 was motivated by many years of experience with developing scientific software, where it was observed that a lot of effort went into developing parts that had little to do with the domain. A significant part of the development process was spent on different software engineering tasks, such as code design, the handling of I/O, correct memory handling of the program state and writing import and export filters in order to use data from different sources. In addition, comes the code maintenance with support of legacy formats and the introduction of new features and changes to internal data state in the scientific software. With SOFT5, it is possible to utilize reusable software components that handle all this, or develop new reusable software components that can be used by others in the same framework.

The main components of SOFT5 is shown in figure 3. The key modules are the tools, storage support and plugin framework. The key tools are the scripting utility and the code generator.

SOFT5 contains a core library with plugin support. The library also comes with an application programming interface to create extensions and custom plugins. The core library is used to connect a software application with the framework.

The main approach to developing software with SOFT5 is to incrementally describe the domain of the software using entities (see below). The entities can represent different

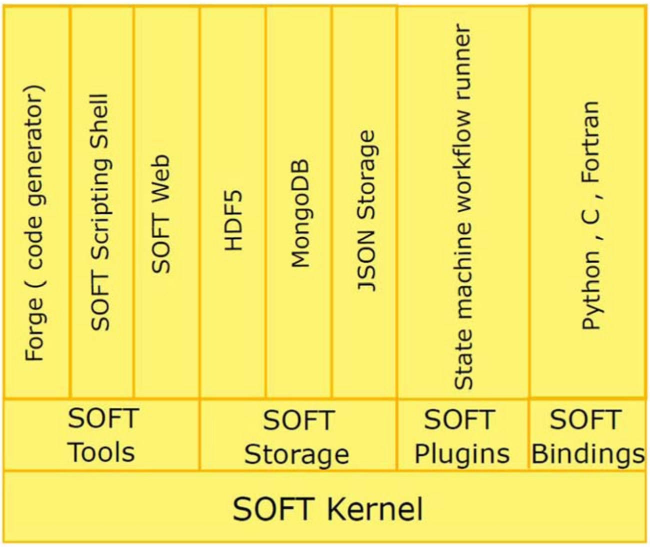


Figure 3. The main components in the SOFT5 platform.

independent elements of the software, and be used in handling I/O as well as in code generation and documentation. Entities can also be used for annotating data and data sets. This might be useful in cases where for instance the origin of the data, license and ownership are of importance.

Since any complex software will have many entities and often multiple instances of the same entity, SOFT5 allows for creating collections of instances with defined relationships between them. These entity collections are called ‘collections’ (see below).

An approach of SOFT5 is that software may be written is such a way that business logic is handled by the codebase, while I/O, file-formats, version handling, data import/export and interoperability can be handled by reusable components in the SOFT5-framework, thus reducing risk and development time.

2.4.1. Entities. An entity can be a single thing or object that represents something physical or nonphysical, concretely or abstract. The entity contains information about the data that constitutes the state of thing it describes. The entity does not contain the actual data, but describes what the different data fields are, in terms of name, data types, units, dimensionality etc. It can be described as formalised metadata that enables for the correct interpretation of a set of data. Hence the entities should be made available together with the software or application in which they are used.

An example of an entity is ‘atom’, which can be defined as something that has a position, an atomic number (which characterizes the chemical element), mass, charge, etc. Another example of a completely different kind of entity can be a data reference-entity with properties such as name, description, license, access-url, media-type, format, etc. The first entity is suitable as an object in a simulation code, while the latter is more suitable for a data catalogue distribution description (see dcat:Distribution)[7]. Entities allow now to describe dedicated

[7] DCAT: Data Catalogue Vocabulary.

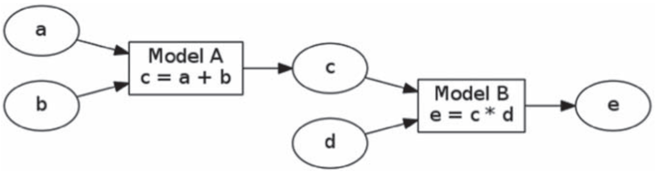


Figure 4. A very simple workflow that connects the output of model A to the input of model B.

aspects of the domain. While each entity describes a single unit of information, a collection of entities can describe the complete domain (check collections below).

Each published entity needs to be uniquely identified in order to avoid confusion. The entity identifier has therefore three separate elements: a name, a namespace and a version number. An entity named ‘particle’ is unlikely to have the same meaning and the set of parameters across all domains. In particle physics, the entity ‘particle’ would constitute matter and charge, while in other fields the term ‘particle’ can be a general term to describe something small. For this reason the SOFT5 entities have namespaces, similar to how vocabularies are defined in OWL[8]. The version number is a pragmatic solution to handle how properties of an entity might evolve during the development process. In order to handle different versions of a software, the entity version number can be used to identify the necessary transformation between two data sets.

2.4.2. Collections. A collection is a container whose instances holds references to a set of entity (or collection) instances and relations between them. Relations are triplets of the form (subject, predicate, object), where subject and object both labels an instance and predicate describes a relation between the subject and object, like ‘child-of’, ‘connected-to’, ‘reactant-of’, etc. This is much more general than hierarchical structures and allow to represent the knowledge of the domain where the data exists. Hence, a collection could also have been called a *context*. It also opens possibilities like deductive databases.

Collections are useful to represent the knowledge of the domain, to find data that relates to other data or to uniquely identify a complete data set with a single identifier.

2.4.3. Storage. A *storage* is a module that can transfer a given state to an external medium (file or database). It is an important abstraction in SOFT5 which allows to transparently handle input to or output from modelling software. This is realised via a plugin-system of drivers (or backends) that translate to and from the external formats. A storage backend may either be generic or specific. A specific storage backend is tied to a given entity and could e.g. read experimental data from a specific instrument into an instance of the entity or write an instance of the entity to file using a specific format expected by a third party software. SOFT has three generic built-in storages, HDF5, JSON and the MongoDB database, that can read or write instances of any entity [20–22]. Furthermore SOFT5 allows the user to write dedicated plugins for a given entity, typically for reading or writing data from or to a specific file format.

2.5. Workflow technology

We will use the very simple example of a workflow shown in figure 4 to illustrate the basic concepts. We have two models A and B, where the output of A is used as input to B. Circles

[8] OWL—Web Ontology Language.

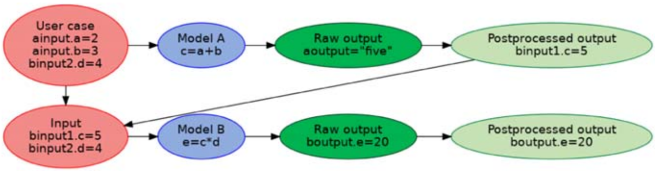


Figure 5. The same workflow as in figure 4 using the MODA template. The translation of the output of model A to the representation expected by model B is performed in postprocessing of the raw output from model A.

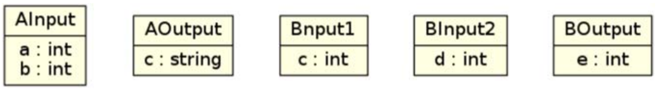


Figure 6. Entities used in this workflow.

represent *physical system state* and boxes *transitions* between different snapshots of the physical system state.

A typical challenge in setting up workflows in real life, is that the output from model A is not represented in the way expected by model B. But, if the output of model A is semantically equivalent to the input needed by model B, it is possible to translate (postprocess) the output of A to the form expected by B and hence establish a robust workflow. Such a case is shown in figure 5, where Model A returns its results as a string 'five' while model B expects a number.

2.5.1. Workflow implementation using SOFT5. To implement this workflow as depicted above, we proceed towards defining the entities as shown in figure 6. The initial state is described by instances of *AInput* and *BInput2* and the final result will be given as an instance of *BOutput*.

A typical and easier way to handle data flow in SOFT, is to let all data flow between models and the system flow via a central storage. A centralized storage not only helps in maintaining an overview of the different tasks, but also allow for easy data backup and version control. To keep track of the instances created for each invocation of the workflow, we create a collection specific to each invocation, in which we store references to all instances belonging to this invocation. The workflow can then be broken down into a set of tasks exposing the same interface. Each task is simply provided a reference to the storage and the collection for the current invocation of the workflow.

Running the workflow involves the following steps:

1. Initialization: Create the collection, populate it with new instances for the initial state defined by the user case and store everything in the storage.

2. Run Model A: This model is implemented as an external program taking the name of a file containing the two integers to be added as the only argument. The result is written to standard output. In order to integrate model A in our workflow we create a wrapper function that performs the following steps:

1. *Pre-process*: Fetch *AInput* and serialize it to a temporary file in the format expected by model A.
2. *Compute*: Execute Model A with the name of the temporary file provided on the command line.

3. *Post-process*: Creates an instance of entity AOutput, populate it with the result of Model A and push it to the common storage. Latter also removes the temporary file.

3. Run Model B: This model is implemented as an external program that uses SOFT. As arguments, it takes references to the storage and the Collection. When called it asks for BInput2 and AOutput, the latter as represented as an instance of BInput1. SOFT will then check if it can find an Translator that can translate an instance of AOutput to an instance of BInput1. When this Translator is found, it is applied. This will create BInput1 from AOutput. The new instance is then handed to model B. Model B then creates an instance of BOutput populated with the product and stores it in the storage.

3. Applying workflow on advanced concrete technology. Bridging density functional theory (DFT) to a continuum corrosion model (CCM) via SOFT5

The above mentioned methodology is applied to the DFT output transfer to a microscale CCM embedded in a multi-scale model consisting of 4 different modelling scales. The aim of this modelling chain is to predict accurately the service life limitation due to chloride ingress of reinforced concrete structures exposed to submerged marine environment. The task consists of bridging four different scales as shown in figure 2. Each scale communicates data with each other. The paper here focuses on data transfer between the first two scales i.e. atomistic/electronic scale to CCM at microscale via SOFT5.

3.1. M3 framework on service life prediction of concrete structures

The proposed workflow involves calculating and up-scaling entities of interest at the atomic/electronic level and their transfer via SOFT5 to the continuum based micro-scale model (CCM). A brief description of the concept and the underlying models is presented.

The atomistic model in this case deals with the flow of corrosion inducing species i.e. chloride ions through a representative cement matrix composed of tobermorite.

The flow of chloride ions through the calcium silicate hydrate (CSH) matrix is considered to be the rate determining step for chloride ingress in cement. The flow through the pore network is CSH will occur as a significantly faster rate than through CSH itself given a non-continuous pore network. Further, the swelling of CSH is also an up-scaling input parameter required by the CCM.The binding of chloride to the cement paste is a well known phenomena and its greatly attributed for extending the time to corrosion initiation as it slows down the transport of chloride ions. The hydration of cement leads to the formation of CSH as a major volumetric phase among many others. CSH is responsible for the majority of chloride binding in concrete [23].

The structure and volume correlations of CSH depending on water content in addition to the interaction of Cl in CSH are investigated from first principles, and is presented in Svenum *et al* [24]. The calculations are performed using the DFT calculations at the PBE-GGA level [25] using the Vienna *ab-initio* simulation package (VASP) [26, 27]. The model takes as input an approximate atomic structure found by manipulating the theoretical model of tobermorite to a correct Ca/Si ratio with the desired water content. The structure and volume of the model is then minimized, and the minimum energy, volume and atomic positions are the output from the simulations. To investigate the chloride diffusion, transition state theory calculations using nudge elastic band is performed [28]. The input for these calculations are the minimized start and end atomic structures. The energy barrier to go from the start structure to the end structure is calculated, in addition to the reaction coordinate. The diffusivity coefficient can be calculated from the energy barriers to move an atom through the periodic cell by

$$D = D_0 \exp\left(-\frac{E_a}{k_B T}\right), \tag{1}$$

where D_0 is the pre-exponential factor, E_a is the activation barrier for diffusion, k_B is the Boltzmann constant and T is the temperature. The pre-exponential factor D_0 can be obtained either by transition state theory [29], or found experimentally [30]. The diffusion should be equal in all directions within the interlayer, while it can be assumed to be zero between the interlayers. This results in a diffusion of D in two directions and 0 in one. Since the microstructure is not considered explicitly in this framework, a simple averaging is performed and the diffusion tensor is estimated to be $D_{ii} = 2/3D$.

On the other hand, the CCM model is a FEM based model implemented in COMSOL multiphysics software which models flow of free chlorides in the pore network and also accounts for bound chlorides in the mass balance. Corrosion processes occurring due to chloride induced breakdown of passive film (native oxide layer) [31, 32] are also considered.

Nernst–Planck equation is used to model transport of chloride ions inside concrete (equation (2)) which is exposed to 3% NaCl . Here Cl represents the concentration of free chloride ions inside the pore network, ϕ_p represents the average concrete porosity, D_{eff} represents the effective diffusion coefficient for flow of Cl in pore network, z_i is the valency number, F represents Faraday constant, R is the ideal gas constant, T is the absolute temperature and φ_{el} is the electic potential. $\frac{\partial \text{Cl}_b}{\partial t}$ represents the rate of bound chlorides Cl_b acting as a sink term in the mass balance (equation (2)). A distribution of electric potential is obtained from Laplace equation (equation (3)) and consequently corrosion currents can be derived, as presented in [33, 34].

$$\phi_p \frac{\partial \text{Cl}}{\partial t} = -\nabla . \left[-D_{\text{eff}} \left(\nabla \text{Cl} + \text{Cl} \frac{z_i F}{RT} \nabla \varphi_{\text{el}} \right) \right] - \frac{\partial \text{Cl}_b}{\partial t} \tag{2}$$

$$\nabla^2 \varphi_{\text{el}} = 0. \tag{3}$$

The rate of chloride binding $\frac{\partial \text{Cl}_b}{\partial t}$ can be related to the diffusivity of Cl ion in the CSH derived from electronic scale model which tends to be a very slow process, wherein CSH entraps Cl ion [24]. Data verification, validation as well as physical relevance of up-scaling of data between scales is an on-going process at this stage. To get a more accurate estimate of the diffusion coefficient in the future, it will be necessary to add a scale which take the microstructure of CSH into account. The framework is well suited to adding this extension.

3.2. Data storage and transfer

The presented workflow is shown in figure 7 and focuses on the data handing between Electronic scale model and the CCM. The Electronic scale model starts with an appropriate atomistic structure of Tobermorite with different calcium to silica ratios as well as different water to Silica ratio. This forms the basic inputs of the Electronic scale model. This model used DFT for performing calculations related to favourable positions of chloride binding and associated diffusion energy of the chloride ion inside the tobermorite matrix.

The computed outputs details on the swelling of the matrix as well as density change of the cementitious matrix. Furthermore atom fraction of water, diffusion corrections for chloride-ion transport and corresponding activation energies are also computed. These listed inputs are directly used an inputs in other models along the modelling chain.

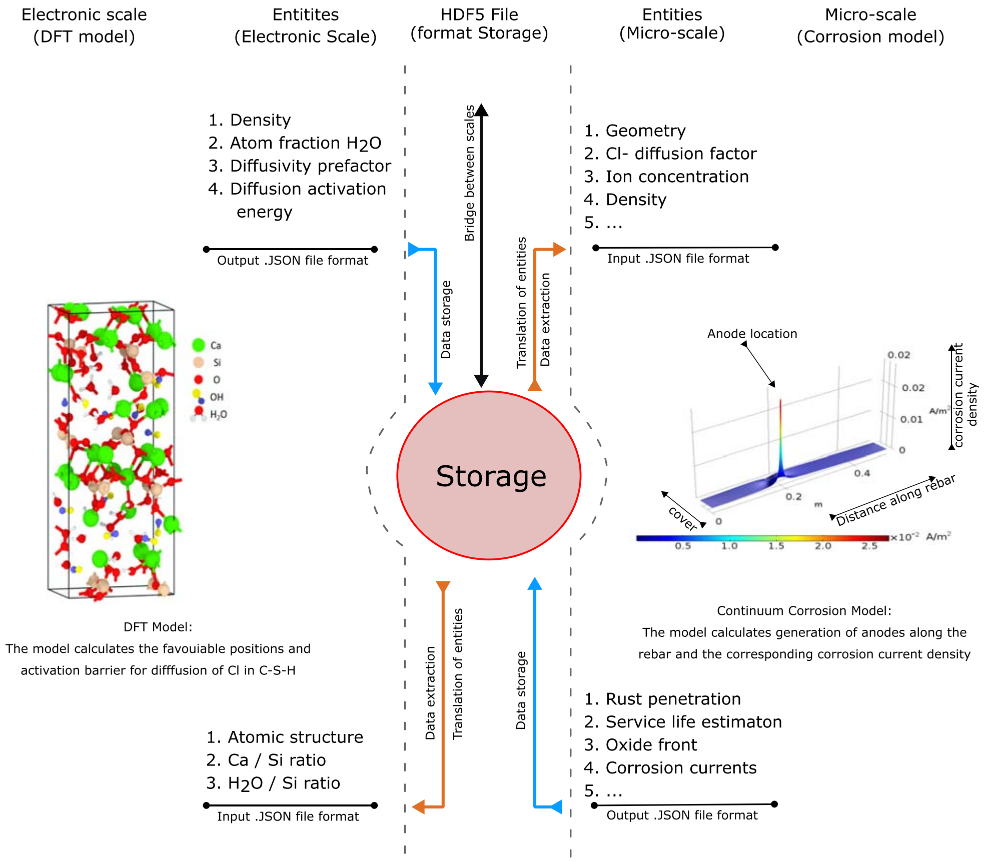


Figure 7. Workflow and data transfer entities between DFT model and continuum corrosion model. The output from .JSON (such as rust penetration, corrosion currents etc in CCM) can be derived from material relations on data in each model.

The CCM can therefore rely on information related to chloride-ion diffusivity and chloride ion binding from the Electronic model in the proposed framework. Together with the geometry of the structural member and other related input factors, the CCM calculates the rust penetration on the rebar surface. The oxide front is calculated likewise as a consequence of corrosion currents and an estimation can be drawn on the service life of the concrete structure.

The input/output of data and entities for each respective model is handled using .json files representing the entities and associated metadata. The listing below shows the entity describing the input to the micro-scale corrosion model.

```
{
  "name": "Ion",
  "version": "0.1",
  "namespace": "https://powerfolder.hzg.de/Lorcenis/meta",
  "description": "Input to micro-scale corrosion model."
  "dimensions": [
    {
    "name": "ncoords",
    "description": "Number of spatial coordinates"
    }
  ],
  "properties": [
    {
      "name": "ion_diffusion",
      "type": "double",
      "dims": ["ncoords", "ncoords"],
      "unit": "m^2/s",
      "description": "Diffusion tensor for ions."
    },
    {
      "name": "concentration",
      "type": "double",
      "unit": "mol/m^3",
      "description": "Ion concentration in bulk."
    },
    {
      "name": "density",
      "type": "double",
      "unit": "kg/m^3",
      "description": "Bulk density."
      }
      {
      "name": "geometry",
      "type": "double",
      "dims": ["nnodes", "ncoords"],
      "unit": "cm",
      "description": "Coordinate"
     }
    ]
  }
```

The name, version and namespace identifies the entity uniquely. The descriptions are meant to be read by humans. One dimension is defined, the number of spatial coordinates in the diffusion tensor (which is always 3). The properties that follows, describes the data records that we want to transfer. Separate instances of four entities are shown in figure 7.

3.3. Application

The above mentioned simulation framework employing SOFT5 shows the workflow of the simulation action and data exchange within the multiscale corrosion modelling framework between the DFT model and the CCM as depicted in figure 7. The atomic supercell represents a typical cement paste CSH structure containing defined arrangement of relevant ions close to the concrete reinforcement. The model allows to calculate relevant entities based on dedicated model input (from the literature) is introduced as well. All data is stored (thereby belonging to a semantic ontology) and accessible for the microscale model. DFT related data is applied as (translated to) model input. In the next step entities like rust penetration changes can be computed within a set of nonlinear PDE's by finite elements and stored back.

4. Discussion

4.1. Data science

The aim of this work is to demonstrate how interoperability between two software, VASP and a user-defined model in COMSOL Multiphysics, that belongs to two completely different domains can be achieved. For this purpose the SOFT framework was used, which is a framework for describing data in a concise and interoperable way. SOFT is not a semantic framework in the sense that it is tied to an ontology for a specific domain. Instead SOFT allows to represent the knowledge and data of any ontology (via collections). In this sense SOFT can be compared to tools like web ontology language (OWL), but is much closer linked to actual software code. The metadata model in SOFT is by design as simple and minimal as possible, while still being complete enough to allow unambiguous data representation and communication as well as code generation.

The shown MODA template itself was developed by the materials modelling community based on gained experience, and works in that way as well, thereby considering pure modelling data but also transformation information. The red, green and light green nodes represent computational data directly related to the physical state of the system, while the blue nodes represent a transformation. In that way the template might be mature enough for extend unambiguous model description.

4.2. Technical limitations and modelling constraints

Modelling and simulation of corrosion process along the entire length and time scale, especially in concrete environment, is a highly demanding computational problem. Feature extraction from each simulation step thereby keeping accuracy and computational efforts at tolerant level is a huge challenge. Limitations are mainly related to modelling gaps. Data generated at lower scales cannot be used directly at higher scale, which belongs to scale-bridging issues of described physio-chemical entities. Due to the multispecies nature of the concrete environment and due to the fact that mixing rules cannot be applied straightforward. Calculations of e.g. density properties or diffusion activation energies are still limited to the single phases and predefined species concentration scenarios. Since even the next modelling scale requires tremendous homogenization action, lower scale information cannot be exchanged. A modelling approach for this is lacking or in other words data representing the scenario adequately at electronic level cannot be or better can be very limited transferred and reduced to data applicable in the microscale corrosion model (translation of entities in figure 7).

Another issue relates to the change of the DFT scenario if the system changes according to the microscale corrosion model. The duration of the time step within the transient corrosion model is limited by the fact that the system change due to microscale entities do not induce atomistic structural changes leading to the need of adaptive action within the DFT model. Within the shown example, such feedback is excluded. However, in terms of accuracy a feedback is strongly recommended. Realization is very demanding and not feasible considering computational costs.

Furthermore, regarding data transfer from a fine-scale model to a coarse-scale model, significant work using adaptive sampling approach in a multiscale framework has been carried out by Rouet-Leduc *et al* and Knap *et al* [35, 36]. In both of these works, data is communicated between fine scale and coarse scale models. Here as well, the transferred data may be described using SOFT5 entities as it can offer some advantage with respect to reusability and code maintenance. Specifically, for the fine scale results in Knap *et al* [36], SOFT5 entities may be useful to enable reuse of fine-scale results in other applications.

4.3. Future developments

The multiphysical nature of many of the engineering tasks demands interaction of multiple software and respective data structures covering a wide range of scales and related data. In order to achieve engineering relevance, or in other words reproducible prediction capabilities, the quality respectively complexity of modelling approaches should increase. Contrarily data reduction is requested and computational costs shall be minimized. Complimentary, uncertainty aspects should be considered. The occurring conflict requires balancing of all mentioned aspects which can be achieved by improved:

1. Model.
2. Coarse-graining.
3. Chloride bonding.
4. Uncertainty quantification and propagation.

While interoperability based on a common shared ontology has not been the scope of this paper, it is definitely something that we will see more of in the future (EMMC is for instance working on an ontology for materials science). However, the ideas and concepts discussed in this paper will most likely be very relevant in the future to, as a way to go from an ontological description of an domain to actual classes and data structures in a software model. One could e.g. imagine future tools that, given a user case formulated in the frame of an ontology, generates the involved SOFT entities. One can then use the code generator in SOFT to generate corresponding code (e.g. C++), such that the model developer only need to implement the transformations between these classes while leaving the rest to the framework.

5. Conclusions

Computational data entities which are expressed in a standard form using relevant application specific metadata structures has inter-operable characteristics. This data can be stored in a structured manner in a centralized storage. As such results, this leads to an easier development and management of complex simulation workflows.

Multiscale modelling approaches for concrete technology and related service life prediction demand huge methodical capabilities of software interaction. Thereby multiphysical tasks, data aspects, but also feature extraction define hard constraints for a modelling and simulation environment. For the shown example of a submodel interaction within a service-life prediction related framework for damage initiation due to chloride ingress, SOFT5 platform has shown to be a useful data science tool. Data Interaction at different scales for different computational methods (discrete versus continuum) was realized.

Interoperability between two models was achieved via a meta-data scheme implemented within the SOFT environment. DFT data output structure was reorganized in a way to enable interaction with the continuum simulation. Surprisingly, despite moderate efforts, the inhibition threshold to use the SOFT concept was not the limiting aspect. There was an issue to make people getting started to use the tool. It seems that the level of abstractions is hindering fast adaption of software like this.

Access to proper databases without any bandwidth limitations was found to be crucial for the concept realization since it is needed for concurrent read/write to the storage. This limiting problem was solved in SOFT by the mongodb backend. Thus, wise definition of entities towards reduction of the amount of concurrent data transferred is highly recommended.

For future applications within highly complex coupled multiscale simulation frameworks interacting on different scales in time and space, a robust and mature metadata structure allowing the interaction of a variety of data formats is required. SOFT and its metadata abstraction enable this computational progress in materials modelling. Based on modelling data (MODA) schemes the transfer of modelling and simulation chains into a working data interaction environment is simplified. However lacking model and data input availability will still limit respective modelling frameworks. Experiments, accuracy studies and related post-processing will still remain an essential part of the working effort.

Acknowledgments

The authors wish to thank European Union's LORCENIS project, European Materials Modelling Council (EMMC) and SIGMA2 for computational resources. The authors acknowledge that part or parts of figure 7 have been taken from the publication 'Enhanced Predictive Modelling of Steel Corrosion in Concrete in Submerged Zone Based on a Dynamic Activation Approach' authored by Mir *et al* (2019) and published in International Journal of Concrete Structures and Materials. The content has been used with modifications solely for academic purpose under Creative Commons License 4.0 (http://creativecommons.org/licenses/by/4.0/).

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

ZMM, JF, and DH proposed the main concept behind this work. This research is a part ZMM's Doctoral studies and parts of this work will be included in his PhD thesis. JF and TFH did the software implementation. ZMM and JF worked on the software application, debugging and associated data transfer. ZMM, JF, TFH, IHS, IGR, NK, and DH co-wrote the manuscript. JF, MLZ and DH supervised the work and gave critical comments. All authors read and reviewed the manuscript

Funding

This research has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No. 685445 (LORCENIS—Long Lasting Reinforced Concrete for Energy infrastructure under Severe Operating Conditions).

Author information

Authors qualifications, current positions, institution address and contact details are included on page 1.

Data availability

Some of the raw/processed data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study.

ORCID iDs

Zahid M Mir https://orcid.org/0000-0001-6488-1167
Inga G Ringdalen https://orcid.org/0000-0001-9844-4352

References

[1] TIME Magazine USA 2019 The Genius Innovation That Made the Great Library of Alexandria Work https://time.com/4730810/first-card-catalog/
[2] Foster G 1966 Technical report International Standard Book Numbering (ISBN)https://web.archive.org/web/20110430024722/http://www.informaticsdevelopmentinstitute.net/isbn.html
[3] NASA 1999 Mars Climate Orbiter Failure Board Releases Report, Numerous Nasa Actions Underway in Response https://mars.jpl.nasa.gov/msp98/news/mco991110.html
[4] LORCENIS 2019 Long Lasting Reinforced Concrete for Energy Infrastructure under Severe Operating Conditions https://www.sintef.no/lorcenis
[5] CEN Workshop Agreement 2018 *Materials modelling - Terminology, classification and metadata* (CWA) 17284
[6] de Bass A F 2017 *What makes a material function?*
[7] Rubin D L *et al* 2006 National center for biomedical ontology: advancing biomedicine through structured organization of scientific knowledge *OMICS: J. Integr. Biol.* **12** 185–98
[8] Ashino T and Fujita M 2006 Definition of web ontology for design-oriented material selection *Data Sci. J.* **5** 52–63
[9] CAPE OPEN LABORATORIES NETWORK 2019 Computer-Aided Process Engineering http://www.colan.org/
[10] Gene Ontology Consortium 2019 The Gene Ontology Project http://geneontology.org/
[11] Ashino T 2010 Materials ontology: an infrastructure for exchanging materials information and knowledge *Data Sci. J.* **9** 54–61
[12] Chalk S J 2016 SciData: a data model and ontology for semantic representation of scientific data *J. Cheminf.* **8** 54
[13] Hey T, Tansley S and Tolle K 2009 *The Fourth Paradigm: Data-intensive Scientific Discovery* (Redmond, WA: Microsoft Research)
[14] Rust P M 2010 PP1 0.1—What is Scientific Data? https://blogs.ch.cam.ac.uk/pmr/2010/07/25/pp01-what-is-scientific-data/
[15] Cheung K, Drennan J and Hunter J 2008 Towards an ontology for data-driven discovery of new materials *AAAI Spring Symp.: Semantic Scientific Knowledge Integration* 9–14
[16] Hunt W H Jr. 2006 Materials informatics: growing from the bio world *JOM* **58** 88
[17] Ojo A, Janowski T and Estevez E 2009 Semantic interoperability architecture for electronic government *Proc. 10th Annual Int. Conf. on Digital Government Research: Social Networks: Making Connections between Citizens, Data and Government* (Digital Government Society of North America) pp 63–72
[18] Schmitz G J, Böttger B, Apel M, Eiken J, Laschet G, Altenfeld R, Berger R, Boussinot G and Viardin A 2016 Towards a metadata scheme for the description of materials—the description of microstructures *Sci. Technol. Adv. Mater.* **17** 410–30
[19] Hagelien T F, Chesnokov A and Johansen S T 2017 A framework for semantic interoperability of scientific software *12th Int. Conf. on CFD in Oil and Gas, Metallurgical and Process Industries (Trondheim, NORWAY)* (SINTEF) pp 317–30
[20] 2017 Standard ECMA-404 The JSON Data Interchange Syntax http://www.ecma-international.org/publications/files/ECMA-ST/ECMA-404.pdf

[21] The HDF Group 2000–2010 Hierarchical Data Format Version 5 http://www.hdfgroup.org/HDF5

[22] Chodorow K 2013 *MongoDB: The Definitive Guide* (Sebastopol, CA: O'Reilly Media, Inc.)

[23] Saetta A V, Scotta R V and Vitaliani R V 1993 Analysis of chloride diffusion into partially saturated concrete *Am. Concr. Inst.—Mater. J.* **90** 441–51

[24] Svenum I-H *et al* 2020 Structure, hydration, and chloride ingress in C-S-H: Insight from DFT calculations *Cem. Concr. Res.* **129** 105965

[25] Perdew J P, Burke K and Ernzerhof M 1996 Generalized gradient approximation made simple *Phys. Rev. Lett.* **77** 3865–8

[26] Kresse G and Furthmüller J 1996 Efficient iterative schemes for *ab initio* total-energy calculations using a plane-wave basis set *Phys. Rev.* B **54** 11169–86

[27] VASP 2018 The Vienna Ab initio Simulation Package (VASP) https://www.vasp.at/about/

[28] Henkelman G, Uberuaga B P and Jónsson H 2000 A climbing image nudged elastic band method for finding saddle points and minimum energy paths *J. Chem. Phys.* **113** 9901–4

[29] Løvvik O M, Sagvolden E and Li Y J 2013 Prediction of solute diffusivity in al assisted by first-principles molecular dynamics *J. Phys.: Condens. Matter* **26** 025403

[30] Lin S H 1993 Chloride diffusion in porous concrete under conditions of variable temperature *Wärme-und Stoffübertragung* **28** 411–5

[31] Saremi M and Mahallati E 2002 A study on chloride-induced depassivation of mild steel in simulated concrete pore solution *Cem. Concr. Res.* **32** 1915–21

[32] Ghods P, Isgor O B, McRae G A, Li J and Gu G P 2011 Microscopic investigation of mill scale and its proposed effect on the variability of chloride-induced depassivation of carbon steel rebar *Corros. Sci.* **53** 946–54

[33] Ožbolt J, Balabanić G and Kušter M 2011 3d numerical modelling of steel corrosion in concrete structures *Corros. Sci.* **53** 4166–77

[34] Höche D 2015 Simulation of corrosion product deposit layer growth on bare magnesium galvanically coupled to aluminum *J. Electrochem. Soc.* **162** C1–11

[35] Rouet-Leduc B *et al* 2014 Spatial adaptive sampling in multiscale simulation *Comput. Phys. Commun.* **185** 1857–64

[36] Knap J, Barton N R, Hornung R D, Arsenlis A, Becker R and Jefferson D R 2008 Adaptive sampling in hierarchical simulation *Int. J. Numer. Methods Eng.* **76** 572–600