D4.1: Report on Application Software Readiness

Due Date	14 March 2019
Delivery	8 March 2019
Submission of updated version	N/A
Lead Partner	MPG
Dissemination Level	Public
Status	Final
Approved	
Version	V1.1

This project has received funding from the *European Union's Horizon 2020 research and innovation programme* under grant agreement No 800925.

DOCUMENT INFO

Date and version number	Author	Comments
25.01.2019 v0.1	Olivier Hoenen	Initial Draft
	Onnie Luk	Add Fusion application
28.01.2019 v0.2	Robin Richardson	Add HemeLB application
	David Wright	Add BAC application
30.01.2019 v0.3	Olivier Hoenen	Add content in sections 2 and 3, fix captions/references
05.02.2019 v0.4	Diana Suleimenova	Add Migration application
	Anna Nikishova	Add ISR application
	Robert Sinclair	Add Materials application
06.02.2019 v0.5	Wouter Edeling	Add Climate application and content in section 3
	Daan Crommelin	
	Dongwei Ye	Add HemoCell application
15.02.2019 v0.6	All	Corrections and improvements
01.03.2019	Bartosz Bosak Erwan Raffin	Internal review
05.03.2019 v1.0	Olivier Hoenen	Corrections after internal review
07.03.2019	Peter Coveney	Provided final review comments
08.03.2019 v1.1	Olivier Hoenen	Final corrections

CONTRIBUTORS

- Olivier Hoenen – Main Author
- Onnie Luk
- David Wright
- Robin Richardson
- Diana Suleimenova
- Anna Nikishova
- Robbie Sinclair
- Wouter Edeling
- Daan Crommelin
- Dongwei Ye

Disclaimer

TABLE OF CONTENTS

LIST OF FIGURES

1 Executive summary

This deliverable reports on achievements from Task 4.1, which was focused on preparing and studying the selected VECMA applications to drive the developments and requirements of the technical work-packages (WP2-3-5). This task was organized along three main axes:

- ensuring that all fast-track applications are ready and available on the initial infrastructure provided by WP5, so they can drive the developments of the VECMA toolkit and serve as early demonstrators for solutions provided within the project,
- reviewing, in collaboration with the technical work-packages, the needs of all VECMA applications in terms of VVUQ algorithms and methods (WP2, will be covered in D2.1), toolkit functionalities and development (WP3, covered in D3.1), and e-infrastructure support (WP5, covered in D5.1),
- starting to plan in more detail the required work for the deep-track tasks.

In this document we show that each of the four applications on the fast-track (*fusion*, *materials*, *Binding-Affinity-Calculator* and *In-Stent-Restenosis*) are running, which means that all their submodels are implemented, together with their corresponding scale-bridging methods, and are using some scripting or coupling software to bring all components together into one high-performance multiscale computing (HPMC) workflow. These fast-track applications have been deployed within the current VECMA e-infrastructure (at time of writing this report, only EAGLE at PSNC was available to the consortium). This is covered, along with the current status of all applications, in the Readiness report section of this deliverable.

A review of all applications needs and requirements in terms of VVUQ methods and dedicated software has been performed via an online questionnaire. Answers to the questionnaire are made available to the consortium, in order to provide a general overview of the applications needs, and can be used at any time by the technical work-packages while developing generic concepts and software. Links to the questionnaire and its results can be found in the Annex of this deliverable.

As an additional effort in driving the developments from technical work-packages towards features that are directly actionable within the applications, each feature specification is provided through an Acceptance Use Case (AUC). Such AUCs are then stored in a GIT repository to keep track of their versions, and allow discussions with WP2 and WP3 developers when more details are required.

Finally, a draft roadmap for the deep-track applications and related developments is given in order to help in identifying and planning major efforts needed for their implementation and integration with the VECMA toolkit.

2 Readiness report

The initial application portfolio in VECMA contains 8 multiscale applications from different domains of great societal importance: fusion, biomedicine, materials science, climate modelling and human migrations. Each of these applications requires systematic validation, verification and uncertainty quantification in order to obtain a necessary level of trust in its outputs so they can deliver actionable results within and outside academia research. The main target of the VECMA project is to develop multiscale oriented but generic VVUQ solutions without, as far as possible, requiring modifications to the underlying application. Nevertheless the presence of these 8 exemplar applications is essential and twofold: (i) driving VECMA's algorithms and software development through an application pull, and (ii) ensuring that these developments are useful, generic (they can be applied to applications in different domains) and could be integrated within existing applications seamlessly. Since software development takes considerable amounts of time, we structured the applications into two tracks: applications for the fast-track toolkit development and functionalities, to provide as early as possible the basic requirements and be used as initial testing targets; and applications for the deep-track to complement initial requirements with more advanced features, including intrusive methods and targeting applications that are not yet fully developed or in production. Of course, applications used for the fast-track may also be used within the deep-track.

2.1 Applications for the fast-track

2.1.1 Fusion

The fusion multiscale application takes the component-based approach in order to incorporate the microscopic effects from turbulence into the study of overall macroscopic plasma transport. It consists of four major components, or *submodels*, which are coupled together following the topology given in **Error! Not a valid bookmark self-reference.**. In the current version, the transport submodel (TRA) is implemented by the ETS module [1], which is a 1D code that evolves temperature and density profiles in time. The plasma equilibrium submodel (EQU) is implemented by CHEASE [2], which is a 2D fixed boundary equilibrium code (Grad-Shafranov solver) that updates geometrical information. The turbulence submodel (TUR) is implemented by GEM [3], a 3D gyrofluid electromagnetic code that determines heat and particle fluxes at each flux-tube in a field-aligned shifted-metric coordinate system. The calculations at each flux-tube are done concurrently. IMP4DV is a module based on dynamical alignment [3], which converts fluxes coming from the turbulence submodel into transport coefficients compatible with the transport submodel. In addition, a simple

component (INIT) imports initial data into the system from various sources (files, databases, etc.). All these codes are tested and implemented in the form of components (or *kernels*) of the coupling library MUSCLE2 [4], which also provide standard *dup* components to scatter data to several destinations. The fusion application topology described has already been fully implemented [5] and tested within the VECMA environment on PSNC's Eagle cluster.

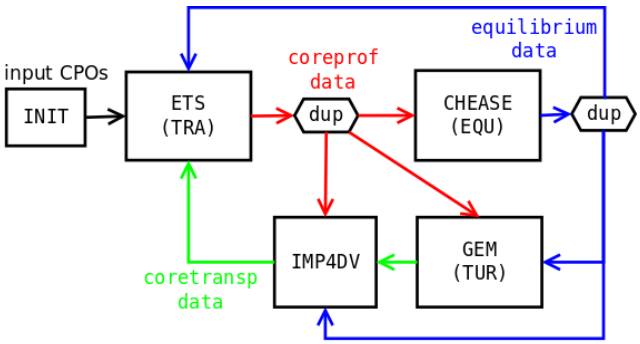

Figure 1: Topology of the fusion application. Each box represents a component of the application, and coloured arrows represent data flow within the application.

The transport coefficients calculated from the turbulence fluxes are inherently noisy. In the fusion workflow, the transport coefficients are propagated through the transport code and produce an uncertainty interval in the calculated profiles, which can then be used in the equilibrium and turbulence codes to calculate new uncertainty intervals. Our ultimate goal is to predict not only the profiles of densities and temperatures, but also the uncertainty in these profiles. This would then allow for a more trustworthy comparison with experimental results, as part of the validation process. To update the workflow so that uncertainties are taken into account, we identified several requirements in terms of VVUQ methods, software/toolkit functionalities, and infrastructure support.

In terms of VVUQ methods, we are currently investigating the use of a polynomial chaos expansion (PCE) method [6]. We begin by treating each submodel separately as a black box, for which statistical metrics are derived directly from the polynomial expansion. However, this method requires a large number of samples on the uncertain input parameters in order to describe the output profile as a distribution with statistical relevance. Due to the Gaussian nature of the probability function omnipresent in the integration, we use a Gauss-Hermite quadrature on 6 points (with a 4^{th} order polynomial to ensure enough accuracy), which for 8 uncertain input parameters of the ETS code, implies that 1.7 million samples need to be run. This highlights the importance of having a smart and scalable Pilot Jobs mechanism as part of the toolkit functionalities. To propagate the uncertainties over to the different elements of the workflow, and for a potentially large number of cycles, we will

need efficient ways to represent such distribution and to perform some sort of reduction. We may also consider different UQ method (for instance quasi-Monte Carlo) for components which have a larger count of uncertain input parameters. Eventually we intend to build a semi-intrusive approach to study how these uncertainties propagate in time and across scales. We will need an e-infrastructure that can provide the MUSCLE2 coupling environment and several third-party libraries and software for the single-scale components, such as FFTW3, LAPACK, BLAS. Use of container technologies would also be of interest as we expand our UQ studies.

2.1.2 Materials

Computational modelling and simulation has become increasingly common in materials research, assisting experimental laboratory work via quantitative predictions [7], economizing time and resources by steering the design of new materials with particular desired properties. This requires assessing mechanical properties at several characteristic length and time scales. The structure of polymer-based materials, including polymer nanocomposites (PNCs), is designed at the nanoscale. PNCs have potential applications in engineering based on chemistry and processing [8, 9, 10] due to their large scale performance under extreme loading conditions [11, 12, 13]. Whilst diverse components (including graphene and clays) can be arranged into nanostructures, at present the resultant macroscopic physical properties are mainly determined by experimental trial and error. A key computational challenge is efficient sampling of all scales relevant to understanding how nanostructure gives rise to macroscopic properties [14, 15]. For example, to accurately simulate fracture behaviour in PNCs one needs to capture stress transfer, void formation, and crack propagation with chemical specificity [11, 16].

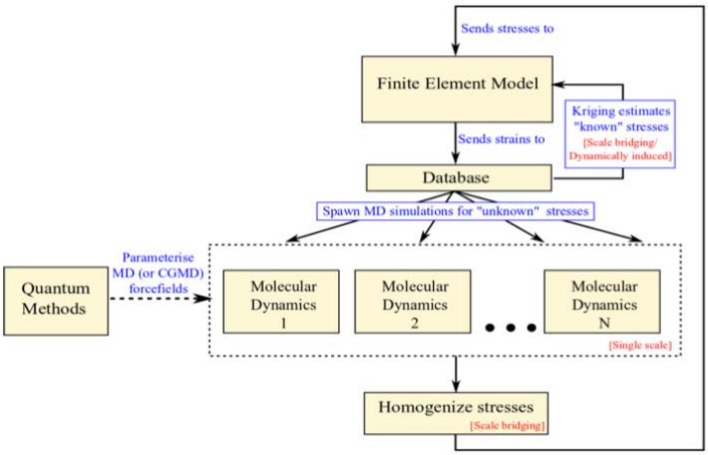

Figure 2: Materials modelling workflow. Density functional theory (DFT) is used to inform molecular dynamics parameterization. Molecular dynamics (MD) and a finite element method (FEM) make up the respective nanoscale and macroscale levels of a coupled simulation. Stress and strain information is passed between the model scales to solve the application problem.

Currently, we have implemented a scheme that couples molecular dynamics (MD) simulations of epoxy-graphene nanocomposites and a finite element method (FEM). The FEM simulations uses Deal_II, a popular finite element solver. It passes the evolution of the strain at discrete points within the engineering scale object to a MD simulation, which uses LAMMPS[1] to then calculate the evolution in strain.

This coupled system is installed, tested and able to execute production runs on the PSNC's machine Eagle. Several UQ needs have been identified, all of which ultimately require more information from the lower scale. To reduce the cost of this problem we must reduce the number of redundant calculations by implementing a database of known calculations from points with 'similar' strain histories, as shown in Figure 2. The simulation is cyclic (for a given stress distribution, the macromodel passes the resulting strain to the micromodel(s) which return the resulting stress distribution to the macromodel, and so on.), and both scale-bridging steps in the cycle will have corresponding uncertainty distributions. This may cause divergence in the distributions over time, due to small variations in the micromodel causing large differences in the macromodel trajectory over time. This represents an example of semi-intrusive cyclic UQ and will require software and methods developed in the deep-track of WP2 and WP3. Uncertainty quantification can however be provided for both the single scale models and the overall workflow using fast track tools (such as EasyVVUQ[2]). Execution of these less intrusive methodologies requires the organisation of the vast number of jobs required to sample the relevant phase space. As such it requires large resources that need well developed scheduling procedures in particular Pilot Jobs that will be provided by WP5.

2.1.3 Binding Affinity Calculator

The Binding Affinity Calculator (BAC) [17] is a molecular dynamics (MD) simulation and binding free energy workflow application. The central goal is to quantify the strength of interactions between small molecules and their target proteins. Two main classes of protocol are automated using BAC; ESMACS (enhanced sampling of molecular dynamics with approximation of continuum solvent) and TIES (thermodynamic integration with enhanced sampling) [18]. The basic workflow topology is acyclic and shared by both protocols (as illustrated in Figure 3), with ensemble simulation used in both cases to provide robust results in a rapid turn around time. The difference between the two protocols from a VVUQ perspective comes in the analysis step. In ESMACS snapshots from the MD

[1] https://github.com/lammps/lammps
[2] https://github.com/UCL-CCS/EasyVVUQ

topology are used as the basis of Poisson Boltzmann based solvation computations that can be viewed as an extra coarse-grained modelling step with associated parameters that can influence the results obtained. In TIES the analysis step is a direct integration of outputs from the simulation. All steps of the BAC workflow are already implemented for both protocols and have been tested on the VECMA e-infrastructure.

All BAC protocols are acyclic, with the priority for uncertainty quantification being non-intrusive analysis (identified UQP1 in the proposal). This approach treats the entire workflow as a "black box" and our requirement here is for a generic method to implement bootstrap statistics over the results of an ensemble simulation. Two potential test cases have been identified for this; the influence of long range cut-off distances on ESMACS results and how differences in ligand parameterization at the quantum mechanical level influence the output of TIES. Furthermore, BAC studies typically use a set of experimental results to evaluate the performance of the protocols on a particular protein system. Consequently, we have a VV pattern requirement to aggregate ensemble results for a set of systems simulated using TIES or ESMACS and quantify the level of correlation between them and the benchmark data. None of these requirements are dependent on theoretical developments from WP2. Concrete requirements for generic sampling of continuous (for example cut-offs) and discontinuous (e.g. protein forcefield choices) parameters, bootstrap statistics, data aggregation and comparison have been passed to WP3. Prototypes of the first two functionalities are already available in the EasyVVUQ library. Current TIES and ESMACS protocols rely upon ensembles of independent simulations which can be simulated asynchronously and even across multiple resources. Given that in real drug design applications vast numbers of systems (1,000,000+) may be required, coordinating the potentially distributed jobs becomes a major challenge. As such we also have a requirement for tools such as Pilot Jobs which can be provided by WP5 to simplify this process.

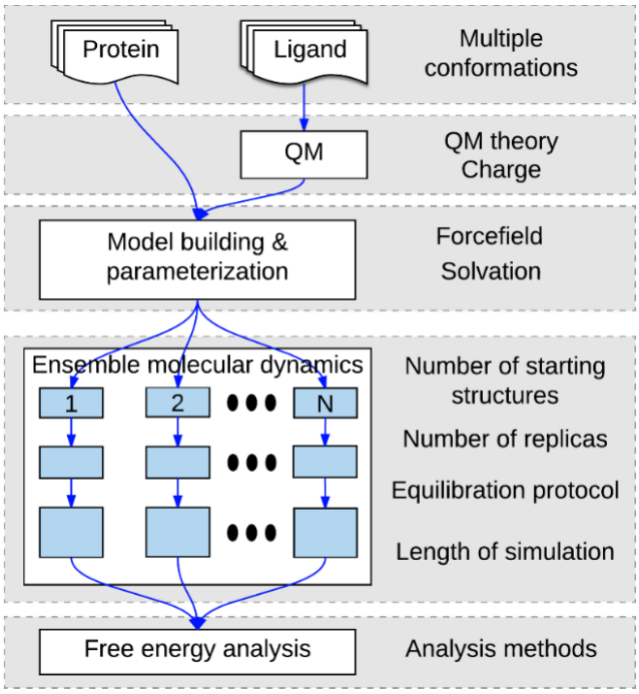

Figure 3: Workflow for the binding affinity calculator (BAC) application. UQ functionality is required for sensitivity analysis of forcefield parameters obtained from the quantum mechanical (QM) theory step and bootstrapping the results of ensemble simulations.

2.1.4 In-Stent Restenosis

The In-Stent Restenosis process [19, 20] can be simulated with a complex 3-dimensional multiscale model called ISR3D [21]. If study using the full 3D model is our final target, in the fast-track we consider the two-dimensional In-Stent restenosis (ISR2D) model. Its domain is a longitudinal section of the artery, in which five subprocesses take place: stent deployment, post-deployment smooth muscle cell proliferation into the lumen, blood flow through the lumen, re-endothelialization, and diffusion of antiproliferative drugs from the stent into the tissue. As these processes take place at different temporal scales, ISR2D is a multiscale model [20, 22, 23], as sketched in Figure 4.

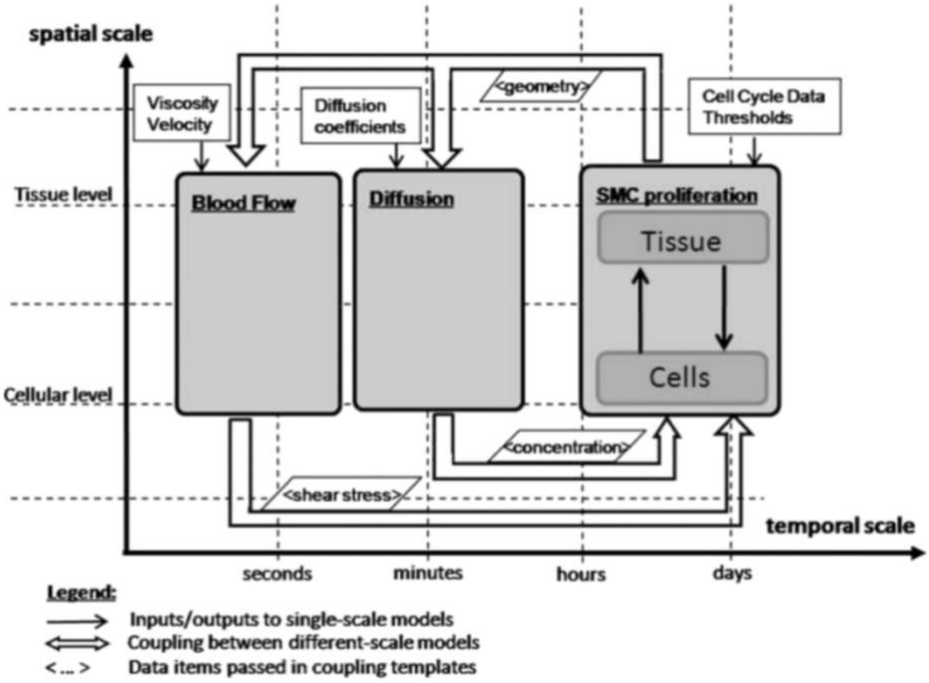

Figure 4: The two-dimensional In-Stent restenosis (ISR2D) model [24]

For the sake of simplicity, drug diffusion has been disabled to simulate a bare metal stent, thus the 2-dimensional model of ISR is reduced to two submodels: a simulation of Smooth Muscle Cell (SMC) growth into the artery lumen, and a simulation of blood flow through the artery (see Figure 5). SMC growth changes the geometry of the artery, which affects the blood flow, while the wall shear stress (WSS) caused by the blood flow inhibits SMC growth when it is above a threshold value. The final output of the model is the cross-sectional area of the neointima, which indicates the degree of restenosis. This model has been deployed on VECMA e-infrastructure, where we will require important HPC resources to run the very large number of instantiations required by the sampling for UQ studies. The identified requirements are the cyclic UQ by the semi-intrusive multiscale metamodeling method. Additionally, the software/toolkit must include sensitivity analysis functionality, since the model is stochastic and the aleatory uncertainty must be always estimated. The requirements for the infrastructure, besides a standard HPC architecture with large amount of storage and low latency network, include MUSCLE2/3 as the coupling framework.

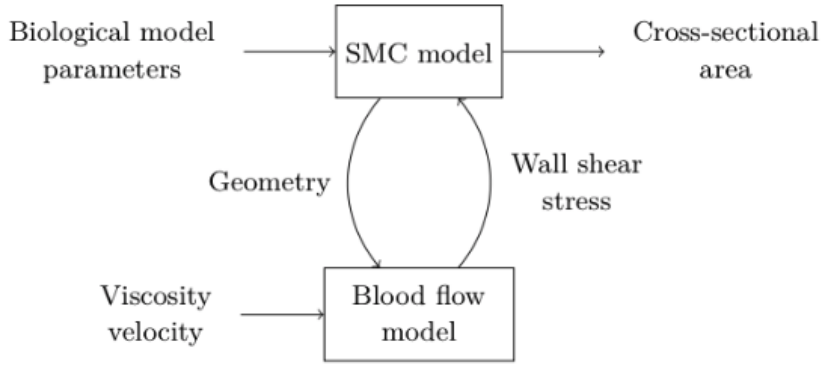

Figure 5: Simplified model of In-stent restenosis [23]

2.2 Applications for the deep-track

2.2.1 Continuum Blood Flow

We use a lattice-Boltzmann solver (HemeLB [25]) to simulate continuum blood flow in large and highly sparse vascular systems in an efficient manner [26]. Through simple acyclic coupling of HemeLB to a 1-D solver, we have been able to study the effects of different patient physiological states, such as heart rate and mean blood pressure, on resulting clinically relevant haemodynamic parameters such as wall shear stress [27]. Magnetic drug targeting simulations have been performed with this model, and some memory efficiency optimizations we have implemented allow far larger and sparser systems than before to be tackled [28]. We recently carried out a validation study of HemeLB simulations of a real patient Middle Cerebral Artery (MCA), using transcranial doppler measurements of the blood velocity profile for comparison, as well as exploring the effects of a change in rheology model or inlet flow rate on the results [29]. We intend to continue these validation and sensitivity analyses, but on a firmer, more systematic footing, leveraging the VECMA toolkit for the extra functionality.

The multiscale model we are building now couples cyclically with a full electrocardiovascular simulation of the human heart (ALYA), which we will use to simulate blood flow in the entire human arterial tree, and potentially the venous tree too - data obtained through collaboration with The Foundation for Research on Information Technologies in Society (IT'IS). The submodel development is complete, but the coupling work is still ongoing. We will then require software tools to compare measured flow profiles against existing data sets, such as those profiles predicted by 1-D simulations and (generally sparse) experimental data points. We will want to understand how our results are sensitive to changes in basic parameters such as blood viscosity, resolution, rheology model, and subsequently what uncertainty estimate we may make regarding the quantities of interest

(pressures, velocity field, wall shear stresses etc). Initial coupling is being done using a bespoke MPI application (with a split MPI communicator) but we later expect to engage with WP5 particularly in the area of Pilot Jobs.

2.2.2 Cell-based Blood Flow

Resolving the properties of whole blood flow on the multiple spatial scales present in the human body is a significant challenge for a single blood flow model. We propose a Heterogeneous Multiscale Model (HMM) for three-dimensional (3D) cell-resolved blood flow (HemoCell) [30, 31, 32] in order to address this challenge. The purpose of this model is to resolve whole blood as a suspension of cells with their emerging rheological properties, on the spatial/time scales where continuous models for blood are normally used. A schematic overview of this 3D HMM model is shown in Figure 6.

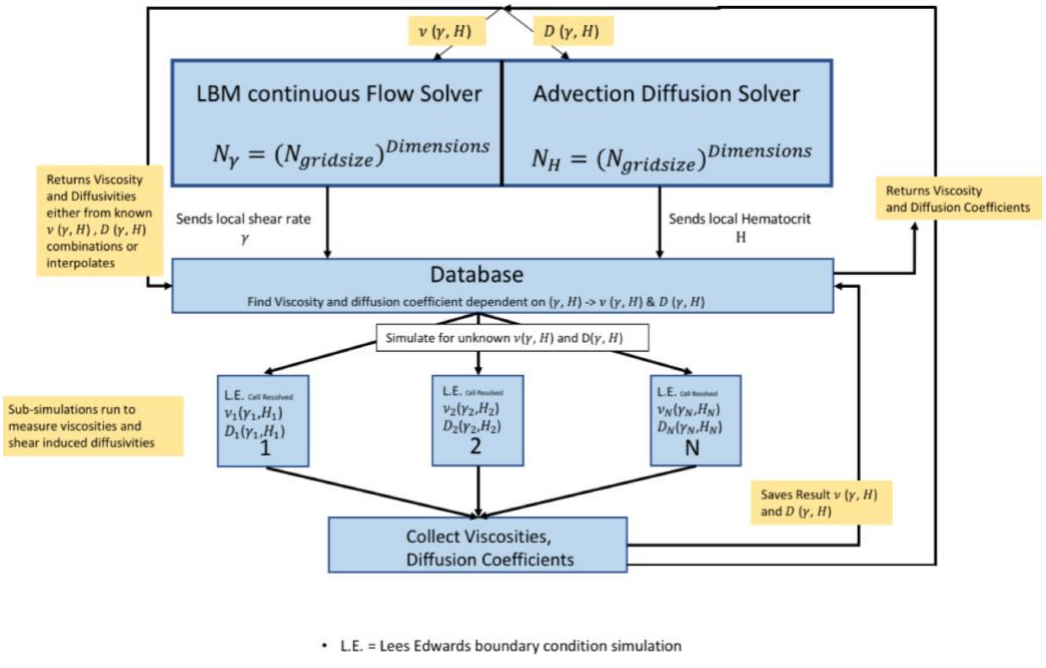

Figure 6: Workflow for the three-dimension cell-resolved blood flow

The blood flow model is a macro-scale continuum flow solver that incorporates a local viscosity which depends on the local cell volume fractions and shear rate. The local cell volume fraction of red blood cells, the haematocrit, is described by an advection–diffusion solver on the macro-scale which requires the shear induced red blood cell diffusivity. This ensures that at every location, a local haematocrit and shear rate will be known. The blood is modelled on the macro-scale as a continuum fluid using the lattice Boltzmann method (LBM) [33, 34]. On the micro-scale, the plasma is modelled by LBM, with a discrete element method for the red blood cells that are coupled to the fluid via the immersed boundary method [35]. The macro-scale advection-diffusion is also modelled with the LBM. The micro-scale models simulate a perfect shear environment using Lees–Edwards boundary conditions for each haematocrit and shear rate combination. These micro-scale models usually occur

in small spatial (~ 100 μ*m*) and temporal scales (~ 1 ms). The micro-scale models send their local viscosities and diffusivities back to the macro-scale model(s) (spatial scale ~ 1 cm and temporal scale ~ 1 second) as input for the lattice velocity calculations which, in turn, are used to calculate shear rate. These velocities and diffusivities are used as input for the advection–diffusion solver to produce the density. Therefore, the micro-scale models measure both the dynamic viscosity and diffusion coefficient of red blood cells in this shear environment and return them to the macro-scale model, which ensures that the local viscosity and the haematocrit profiles on the macro-scale are informed by the cell-resolved models at the micro-scale. The database is used to store data from earlier microscopic simulations to avoid duplicating calculations and to build a surrogate model (e.g. based on a Gaussian random process) to conduct interpolations for similar parameters. This process decreases the required number of micro-scale models requested, which in turn decreases computation time.

We need to understand how sensitive HemoCell is to changes in mechanical parameters of the RBCs, and how this sensitivity propagates to the quantities of interest in the multiscale model. We are preparing to perform such sensitivity analysis, in combination with parameter estimation relying on inverse UQ, starting from experimental data that we are collecting in other projects. These mechanical properties may depend on the individual or on change in physiological conditions. We will therefore perform an extended set of sensitivity analysis and uncertainty quantification on this model, first to explore its behaviour in parameter space, and next to study in detail the uncertainties in specific situations. This deep-track application will require advanced semi-intrusive UQs, where the expensive microscale simulations need to be replaced by surrogates. However, as in the application itself we build surrogates anyway, we should be able to leverage that in terms of performing UQ. This will be part of research for the next phase of the VECMA project. As a single HemoCell run requires anything between a few hours on $O(10^3)$ cores, we expect that a full HMM application would quickly need two orders of magnitude more resources. How this would then translate to a UQ on this application remains to be seen, as this critically depends on the quality of the surrogates that can created.

2.2.3 Climate

Within the climate application we focus on uncertainties stemming from physical and dynamical processes that remain unresolved in numerical models of the global atmosphere or oceans, in particular atmospheric convection and cloud processes. In [36], instances of the Dutch Atmospheric Large Eddy Simulation model (DALES), a high-resolution convection-resolving model on a limited horizontal domain (e.g. 40km x 40km), are nested within selected vertical model columns of the

global atmosphere model OpenIFS. The bi-directional coupling between the DALES instances and OpenIFS is implemented in Python and uses the OMUSE framework [37]. The DALES instances do not communicate directly with each other, only with OpenIFS, hence they can run in parallel. Further developments of the model coupling are ongoing.

The coupled OpenIFS-DALES model can be used to study uncertainties due to convection and clouds, which forms a very challenging application test case for surrogate modelling methods to be developed and implemented in WP2. Notably, for this application, surrogates replacing the DALES instances should be able to reflect the uncertainties stemming from the internal nonlinear and often chaotic dynamics of DALES. This calls for stochastic surrogate models, in line with recent developments on stochastic parameterization in weather and climate modelling; see e.g. [38, 39]. For development purposes, it is useful to have test models available that are a step down in the model complexity hierarchy compared to OpenIFS-DALES. Two such models, both implemented and available to us, are the Lorenz-96 model used in e.g. [40] and the 2-d QG ocean model used in e.g. [41].

For the OpenIFS-DALES model, replacing DALES models by surrogates will require training data which are expensive to generate. For the simulations discussed in [36], 8 cores were used for each DALES instance (360 in total). At this point it is difficult to estimate precisely how many computational resources will be needed to generate a sufficiently large set of training data; however, we project that it might easily involve $O(10^3\text{-}10^4)$ cores for $O(10^2)$ wall clock hours (a hypothetical simulation with a DALES instance in each OpenIFS column would require $O(10^6)$ cores during the same wall clock time). Simulations in [36] were run on a single system (Cray-XC40 at ECMWF); technology for runs on distributed systems has not been developed so far. For this application, the main goal within VECMA is the development, efficient implementation and use of the surrogate modelling methodology.

2.2.4 Multiscale Migration Prediction

Forced population displacement reached record levels in 2017 with 25.4 million refugees spread worldwide. Forecasting forced displacement is vital, as governments and NGOs can save refugee lives and allocate humanitarian resource efficiently [42]. We use an agent-based modelling tool, namely FLEE[3], to forecast the distribution of incoming refugees across destination camps for African countries [43]. To automate the construction, execution and analyses of ensemble simulations of refugee movements [44], we create a FabSim3-based FabFlee plugin. It provides an environment for researchers and organisations to construct and modify refugee simulations, instantiate and execute

[3] https://github.com/djgroen/flee-release

multiple runs for different policy decisions, as well as to validate and visualise the obtained results against the existing data [45].

Within VECMA, we focus on input data uncertainty, sensitivity analysis of simulation runs, verification of our initial model and validation of obtained simulation results. We are in the process of combining FabFlee plugin and the EasyVVUQ toolkit to perform sensitivity analyses for varying agent awareness levels, speed limits of refugee movements and other simulation parameters. It allows us to automate parameter exploration analysis and explore basic one-at-a-time input uncertainty quantification. Importantly, uncertainty quantification and sensitivity analysis are required in multiscale migration application to understand in what regime and scenario our simulation approach performs well. We also intend to use Pilot Jobs functionalities, due to the very large number of anticipated runs.

The multiscale migration workflow has an ensemble of macroscale models and microscale settlement models. The former models run for the full simulation periods, while the latter models spawn when conflicts erupt by exchanging agents in 2-ways (Figure 7). We are in the progress of parallelising Flee [46] with a basic multiscale coupling file, which may undertake the HMM approach.

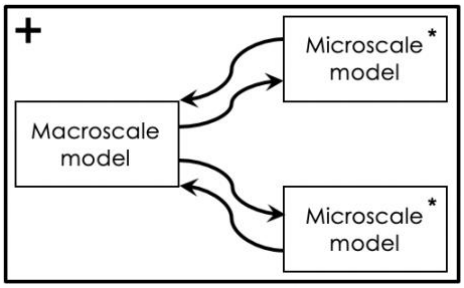

Figure 7: Envisioned multiscale migration workflow

3 Providing inputs for the technical work-packages

In addition to ensuring that all fast-track applications are ready (2.1) and that work has started in the deep-track applications (section 2.2), one of the most fundamental objectives of Task 4.1 was to interact with the three technical work-packages to provide application requirements and agree on feature requests which should be generic enough to be useful to more applications and not implemented in a tailor-made fashion for the initial requester. This was especially challenging during the first months of the project when each work-package was kick-starting its activities at a different pace (due to availability of people over the summer or late hiring), which made it difficult to synchronize early transversal efforts. To allow as much as possible to progress asynchronously in cross-work-packages efforts, we agreed to formalize applications needs through three forms: an

online questionnaire to provide extensive description of each application (3.1), a list of Acceptance Use Cases to provide application-agnostic feature requests (3.2), and a roadmap for deep-track challenges and milestones (3.3).

3.1 Application questionnaire

To gather needs and requirements from the targeted applications we created a questionnaire to guide each developer towards describing their application, which, compared to requesting a full text-based report (for instance given as a summary in section 2) has pros and cons: on the one hand, it is less flexible than an unstructured report, but on the other hand it makes sure that application groups from our five different domains can provide useful information in a uniform and comparable way. It also allows the technical work-packages (WP2, WP3 and WP5) to provide specific questions toward their respective topics of interest (methods and algorithms, toolkit software, and infrastructure). This questionnaire was created collectively as an online form composed of several sections labelled with specific relevant work-package of interest. Some items are covered using free text answers in order to provide flexibility in the descriptions. Answers from all internal applications plus one external application were gathered in November (M5), and results of the analysis performed by WP5 were used to propose a consistent software environment as part of D5.1. A link to the form, its answers and analytics can be found in the Annex.

3.2 Acceptance Use Cases

In addition to generic needs and requirements description recorded in the questionnaire, WP3 and WP2 have expressed the need for more specific feature requests submitted by applications following a use-case-driven approach. Implementing a proposal from WP3 and in line with its software development process (based on evolutionary prototyping, as described in D3.1), the concept of Acceptance Use Case (AUC) has been introduced. An AUC is a piece of script, or text, which describes not only the requested feature with its expected outcome, but also a means to check its implementation validity and usefulness. These checks aim at certifying that implemented software from WP3 and implemented algorithms or methods from WP2 are directly actionable within the applications from WP4. In order to keep track of the changes made among the different, evolving versions of AUCs, and to make them evolve through different versions, we store them inside a specific GIT repository hosted on the project's github[4], which also provides a platform to interact through issues for further analysis and discussion around a single or a group of AUCs. There are

[4] https://github.com/vecma-project

currently 22 AUCs stored in this repository. A templated example of an AUC is given in the VECMA wiki (link available in the Annex).

3.3 Deep-track roadmap

We identify two main types of challenges for the deep-track:

1. The application of non-intrusive methods, implemented in the toolkit (VECMAtk) via, for instance, EasyVVUQ (see D3.1 for more details), to computationally intensive problems (requiring petascale or higher levels of HPC) which were not covered in the fast-track.
2. The implementation of new semi-intrusive and intrusive UQ methodologies (developed under WP2), and their final use in the application grand challenges.

Regarding the first point, we will identify which are the computationally intensive applications, and what are their associated uncertainties. This includes information such as the computational cost of a single run, which inputs are considered uncertain, their assumed probability distributions and the required number of samples. Using the various samplers available in EasyVVUQ (e.g. Monte Carlo, Stochastic Collocation, etc…), one can sample the total input space resulting in a computationally expensive UQ campaign. The main challenge here, tied to WP3 and WP5, is to scale up the tools implemented in EasyVVUQ to petascale or higher.

Regarding the second point, part of WP2 is the development of new UQ methods for multiscale applications. One important component of this development is the deployment of surrogate models, meant to replace (expensive) single-scale models. If the coupling topology is cyclic, information is continually exchanged between the surrogate and the resolved-scale model, thus requiring a dynamic surrogate model. This is in contrast with the fast-track UQ methods, which simply propagate parameter values, without using surrogates.

One possible semi-intrusive UQ method, as proposed in [24], combines the aforementioned speed-up via surrogate modelling with the parametric UQ methods implemented in EasyVVUQ. This allows for the creation of UQ campaigns for the expensive deep-track applications, with a sufficient number of samples in order to obtain converged estimates of the (parametric) output uncertainty.

To facilitate the VECMAtk development towards the deep-track, we are considering intermediate complexity test problems, which are interesting from a multiscale modelling point of view yet fairly inexpensive to run, to allow for the prototyping of new UQ methods. As an example, for the climate application, a suitable intermediate complexity system is the ocean flow model (with runtime of

roughly 12 hours) used in [41]. It is a useful testbed for new dynamic surrogate models of the unresolved scales.

4 Conclusion

In this deliverable, we have presented the status of all eight VECMA applications. All fast-track applications are implemented with their submodels and coupling methods, and can be used to drive and test the early developments from the technical work-packages WP2-3-5. Implementation of the deep-track applications has started and a deep-track roadmap was established to focus on two challenges: the scaling issues for compute intensive non-intrusive methods on the one hand and the more intrusive methods on the other hand. Several means of communication regarding needs and feature requirements were established in close collaboration with the technical work-packages (online form, AUC repository). Initial sets of such requirements, briefly summarized in this deliverable and reflected upon by D3.1, D5.1 and in the future D2.1 deliverables, were already collected and stored for both fast track- and deep-track applications.

Also worth mentioning is the established link with a first potential application external to the VECMA partners, which will be described in forthcoming deliverables and should allow us to improve and demonstrate the versatility and ease of use of future public releases of the toolkit.

5 Annex

Links to the online Application Description Questionnaire, its results and analytics, as well as AUCs template and dedicated GIT repository can be found via the WP4 page of the VECMA wiki:

https://wiki.vecma.eu/vecma-wp4

References

[1] D. Coster, V. Basiuk, G. Pereversez, D. Kalupin, R. Zagorski, R. Stankiewicz, P. Huynh and F. Imbeaux, "The European Transport Solver," *IEEE Trans. Plasma Sci.*, vol. 38, no. 9, pp. 2085-2092, 2010.

[2] H. Lütjens, A. Bondeson and O. Sauter, "The CHEASE code for toroidal MHD equilibria," *Computer Physics Communications,* vol. 97, no. 3, pp. 216-260, 1996.

[3] B. Scott, "Dynamical alignment in three species tokamak edge turbulence," *Physics of Plasmas,* vol. 12, no. 8, p. 082305, 2005.

[4] J. Borgdorff, M. Mamonski, B. Bosak, K. Kurowski, M. Ben Belgacem, B. Chopard, D. Groen, P. Coveney and A. Hoekstra, "Distributed multiscale computing with MUSCLE 2, the Multiscale Coupling Library and Environment," *Journal of Computational Science,* vol. 5, no. 5, pp. 719-731, 2014.

[5] O. Luk, O. Hoenen, A. Bottino, B. Scott and D. Coster, "ComPat Framework for Multiscale Simulations Applied to Fusion Plasmas," *Computer Physics Communications,* 2019.

[6] R. Preuss and U. von Toussaint, "Investigations on Bayesian uncertainty quantification with two examples," in *AIP Conference Proceedings*, 2016.

[7] J. Suter, P. Coveney, R. Anderson, H. Greenwell and S. Cliffe, "Rule based design of clay-swelling inhibitors," *Energy Environ Sci.,* vol. 4, no. 11, pp. 4572-4586, 2011.

[8] P. Lebaron, Z. Wang and T. Pinnavaia, "Polymer-layered silicate nanocomposites: An overview," *Appl Clay Sci.,* vol. 15, no. 1, pp. 11-29, 1999.

[9] M. Bockstaller, R. Mickiewicz and E. Thomas, "Block copolymer nanocomposites: Perspectives for tailored functional materials," *Adv Mater.,* vol. 17, no. 11, pp. 1331-1349, 2005.

[10] S. Ray, "Recent trends and future outlooks in the field of clay-containing polymer nanocomposites," *Macromol Chem Phys.,* vol. 215, no. 12, pp. 1162-1179, 2014.

[11] P. Podsiadlo, A. Kaushik, E. Arruda, A. Waas, B. Shim, J. Xu and et al., "Ultrastrong and Stiff Layered Polymer Nanocomposites," *Science,* vol. 318, no. 5847, pp. 80-3, 2007.

[12] T. Hsieh, A. Kinloch, K. Masania, A. Taylor and S. Sprenger, "The mechanisms and mechanics of the toughening of epoxy polymers modified with silica nanoparticles," *Polymer (Guildf),* vol. 51, no. 26, pp. 6284-6294, 2010.

[13] R. Ritchie, "The conflicts between strength and toughness," *Nat Mater.,* vol. 10, no. 11, pp. 817-822, 2011.

[14] J. Suter, P. Coveney, H. Greenwell and M. Thyveetil, "Large-scale molecular dynamics study of montmorillonite clay: Emergence of undulatory fluctuations and determination of material properties," *J Phys Chem C.,* vol. 111, no. 23, pp. 8246-8259, 2007.

[15] J. Suter, R. Anderson, H. Greenwell and P. Coveney, "Recent advances in large-scale atomistic and coarse-grained molecular dynamics simulation of clay minerals," *J Mater Chem.,* vol. 19, no. 17, p. 2482, 2009.

[16] Y. Zhang and J. Evans, "Approaches to the manufacture of layered nanocomposites," *Appl Surf*

Sci., vol. 258, no. 6, pp. 2098-2102, 2012.

[17] S. Sadiq, D. Wright, S. Watson, S. Zasada, I. Stoica and P. Coveney, "Automated Molecular Simulation Based Binding Affinity Calculator for Ligand-Bound HIV-1 Proteases," *Journal of Chemical Information and Modeling,* vol. 48, no. 9, pp. 1909-1919, 2008.

[18] S. Wan, A. Bhati, S. Zasada, I. Wall, D. Green, P. Bamborough and P. Coveney, "Rapid and Reliable Binding Affinity Prediction of Bromodomain Inhibitors: A Computational Study," *Journal of Chemical Theory and Computation,* vol. 13, no. 2, pp. 784-795, 2017.

[19] H. Tahir, A. Hoekstra, E. Lorenz, P. Lawford, D. Hose, J. Gunn and D. Evans, "ulti-scale simulations of the dynamics of in-stent restenosis: impact of stent deployment and design," *Interface Focus,* vol. 1, no. 3, pp. 365-373, 2011.

[20] D. Evans, P. Lawford, J. Gunn, D. Walker, D. Hose, R. Smallwood, B. Chopard, M. Krafczyk, J. Bernsdorf and A. Hoekstra, "The application of multiscale modelling to the process of development and prevention of stenosis in a stented coronary artery," *Philosophical Transactions of the Royal Society A,* vol. 366, no. 1879, pp. 3343-3360, 2008.

[21] P. Zun, T. Anikina, A. Svitenkov and A. Hoekstra, "A Comparison of Fully-Coupled 3D In-Stent Restenosis Simulations to In-vivo Data," *Front. Physiol.*, vol. 8, p. 284, 2017.

[22] A. Caiazzo, D. Evans, J.-L. Falcone, J. Hegewald, E. Lorenz, B. Stahl, D. Wang, J. Bernsdorf, B. Chopard, J. Gunn, R. Hose, M. Krafczyk, P. Lawford, R. Smallwood, D. Walker and A. Hoekstra, "A complex automata approach for in-stent restenosis: Two- dimensional multiscale modelling and simulations," *Journal of Computational Science,* vol. 2, no. 1, pp. 9-17, 2011.

[23] A. Caiazzo, D. Evans, J.-L. Falcone, J. Hegewald, E. Lorenz, B. Stahl, D. Wang, J. Bernsdorf, B. Chopard, J. Gunn, R. Hose, M. Krafczyk, P. Lawford, R. Smallwood, D. Walker and A. Hoekstra, "Towards a Complex Automata Multiscale Model of In-Stent Restenosis," in *International Conference on Computational Science (ICCS 2009)*, Baton Rouge, 2009.

[24] A. Nikishova, L. Veen, P. Zun and A. Hoekstra, "Semi-intrusive multiscale metamodelling uncertainty quantification with application to a model of in-stent restenosis," *Phil. Trans. R. Soc. A*, vol. 377, p. 20180154, 2019.

[25] M. Mazzeo and P. Coveney, "HemeLB: A high performance parallel lattice-Boltzmann code for large scale fluid flow in complex geometries," *Computer Physics Communications,* vol. 178, no. 12, pp. 894-914, 2008.

[26] D. Groen, J. Hetherington, H. Carver, R. Nash, M. Bernabeu and P. Coveney, "Analysing and modelling the performance of the HemeLB lattice-Boltzmann simulation environment," *Journal of Computational Science,* vol. 4, pp. 412-422, 2013.

[27] M. Itani, U. Schiller, S. Schmieschek, J. Hetherington, M. Bernabeu, H. Chandrashekar, F.

Robertson and P. Coveney, "An automated multiscale ensemble simulation approach for vascular blood flow," *Journal of Computational Science,* vol. 9, pp. 150-155, 2015.

[28] A. Patronis, R. Richardson, S. Schmieschek, B. Wylie, R. Nash and P. Coveney, "Modeling Patient-Specific Magnetic Drug Targeting Within the Intracranial Vasculature," *Frontiers in physiology,* vol. 9, p. 331, 2018.

[29] D. Groen, R. Richardson, R. Coy, U. Schiller, H. Chandrashekar, F. Robertson and P. Coveney, "Validation of Patient-Specific Cerebral Blood Flow Simulation Using Transcranial," *Frontiers in physiology,* p. 721, 2018.

[30] G. Závodszky and et al., "Hemocell: a high-performance microscopic cellular library," *Procedia Computer Science,* vol. 108, pp. 159-165, 2017.

[31] S. Alowayyed and et al., "Load balancing of parallel cell-based blood flow simulations," *Journal of computational science,* vol. 24, pp. 1-7, 2018.

[32] B. Chopard and et al., "A lattice Boltzmann modeling of bloodflow in cerebral aneurysms," in *V Eur Conf Comput Fluid Dyn ECCOMAS CFD*, 2010.

[33] L. Mountrakis and et al., "Parallel performance of an IB-LBM suspension simulation framework," *Journal of computational scienc,* vol. 9, pp. 45-50, 2015.

[34] L. Mountrakis, E. Lorenz and A. Hoekstra, "Validation of an efficient two-dimensional model for dense suspensions of red blood cell," *International Journal of Modern Physics C,* vol. 25, no. 12, p. 144100, 2014.

[35] G. Závodszky and et al., "Cellular level in-silico modeling of blood rheology with an improved material model for red blood cell," *Frontiers in physiolog,* vol. 8, p. 563, 2017.

[36] F. Jansson, G. van den Oord, I. Pelupessy, J. Grönqvist, A. Siebesma and D. Crommelin, "Regional superparameterization in a global circulation model using Large Eddy Simulations," *submitted ,* 2018.

[37] I. Pelupessy, B. van Werkhoven, A. van Elteren, J. Viebahn, A. Candy, S. Portegies Zwart and H. Dijkstra, "The Oceanographic Multipurpose Software Environment (OMUSEv1.0)," *Geoscientific Model Development,* vol. 10, no. 8, pp. 3167-3187, 2017.

[38] J. Dorrestijn, D. Crommelin, A. Siebesma and H. Jonker, "Stochastic convection parameterization with Markov chains in an intermediate complexity GCM," *J. Atmos. Sci.,* vol. 73, pp. 1367-1382, 2016.

[39] J. Berner and et al., "Stochastic Parameterization: Towards a new view of Weather and Climate Models," *Bull. Amer. Meteor. Soc.,* vol. 98, pp. 565-588, 2017.

[40] D. Crommelin and E. Vanden-Eijnden, "Subgrid scale parameterization with conditional Markov chains," *J. Atmos. Sci.,* vol. 65, pp. 2661-2675, 2008.

[41] W. Verkley, P. Kalverla and C. Severijns, "A maximum entropy approach to the parametrization of subgrid processes in two-dimensional flow," *Quarterly Journal of the Royal Meteorological Society*, vol. 142, no. 699, pp. 2273-2283, 2016.

[42] D. Groen, "Simulating Refugee Movements: Where would You Go?," *Procedia Computer Science*, vol. 80, pp. 2251-2255, 2016.

[43] D. Suleimenova, D. Bell and D. Groen, "A generalized simulation development approach for predicting refugee destinations," *Scientific Reports,* vol. 7, no. 1, p. 13377, 2017.

[44] D. Suleimenova, D. Bell and D. Groen, "Towards an automated framework for agent-based simulation of refugee movements," in *50th Winter Simulation Conference (WSC17)*, Las Vegas, 2017.

[45] D. Suleimenova and D. Groen, "How policy decisions affect refugee journeys in South Sudan: A study using automated ensemble simulations," *Journal of Artificial Societies and Social Simulation (JASSS)*, (submitted).

[46] D. Groen, "Development of a Multiscale Simulation Approach for Forced Migration," in *International Conference on Computational Science (ICCS 2018)*, Wuxi, 2018.