DataStorm-FE: A Data- and Decision-Flow and Coordination Engine for Coupled Simulation Ensembles*

Hans Walter Behrens
hwb@asu.edu

K. Selçuk Candan
candan@asu.edu

Xilun Chen
xilun.chen@asu.edu

Ashish Gadkari
agadkari@asu.edu

Yash Garg
ygarg@asu.edu

Mao-Lin Li
mao-lin.li@asu.edu

Arizona State University,
Tempe, AZ 85281-3677

ABSTRACT

Data- and model-driven computer simulations are increasingly critical in many application domains. Yet, several critical data challenges remain in obtaining and leveraging simulations in decision making. Simulations may track 100s of parameters, spanning multiple layers and spatial-temporal frames, affected by complex inter-dependent dynamic processes. Moreover, due to the large numbers of unknowns, decision makers usually need to generate ensembles of stochastic realizations, requiring 10s-1000s of individual simulation instances. The situation on the ground evolves unpredictably, requiring continuously adaptive simulation ensembles. We introduce the `DataStorm` framework for simulation ensemble management, and demonstrate its `DataStorm-FE` data- and decision-flow and coordination engine for creating and maintaining coupled, multi-model simulation ensembles. `DataStorm-FE` enables end-to-end ensemble planning and optimization, including parameter-space sampling, output aggregation and alignment, and state and provenance data management, to improve the overall simulation process. It also aims to work efficiently, producing results while working within a limited simulation budget, and incorporates a multivariate, spatiotemporal data browser to empower decision-making based on these improved results.

PVLDB Reference Format:
H. W. Behrens *et al.* DataStorm-FE: A Data- and Decision-Flow and Coordination Engine for Coupled Simulation Ensembles. *PVLDB*, 11 (12): 1906-1909, 2018.
DOI: https://doi.org/10.14778/3229863.3236221

*Authors are listed in alphabetical order. Research is supported by NSF#1318788 "Data Management for Real-Time Data Driven Epidemic Spread Simulations", NSF#1339835 "E-SDMS: Energy Simulation Data Management System Software", NSF#1610282 "DataStorm: A Data Enabled System for End-to-End Disaster Planning and Response", NSF#1633381 "BIGDATA: Discovering Context-Sensitive Impact in Complex Systems", and "FourCmodeling": EU-H2020 Marie Sklodowska-Curie grant agreement No 690817.

Proceedings of the VLDB Endowment, Vol. 11, No. 12

DOI: https://doi.org/10.14778/3229863.3236221

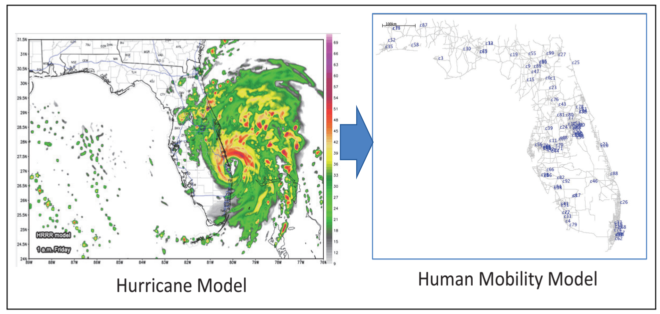

Figure 1: Coupled simulation of a hurricane and human mobility

1. INTRODUCTION

Data- and model-driven computer simulations are increasingly critical in many application domains [12, 11, 14, 6, 13, 7]. For example, when predicting the evolution of epidemics and assessing the impact of interventions, experts often rely on epidemic models and simulation software, such as GLEaM [8] and STEM [3], and simulation ensemble tools, such as EpiDMS [12]. Similarly, data-driven computer simulations for disaster preparedness and response can play a key role in predicting the evolution of disasters and effectively managing emergencies through intervention measures [1].

1.1 Data Challenges in Simulation Ensembles

Yet, several critical data challenges remain in obtaining and leveraging simulations in decision making. Disaster simulations, for example, need to track 100s of inter-dependent parameters, spanning multiple models and geo-spatial frames, affected by complex inter-dependent dynamic processes operating at different resolutions (Figure 1). This is a major challenge due to overlapping and cascading processes, especially when involving multi-hazard scenarios where one hazard (e.g. flooding) is the gateway to the next (e.g. an epidemic). Yet, today's *silo-based, de-coupled simulation engines* assume that disaster, population dynamics, transportation, and disease/epidemic simulations are not integrated, failing to provide an end-to-end view of the disaster and preventing timely and informed decision making.

Moreover, due to the large number of unknowns, decision makers usually need to generate ensembles of stochastic scenarios, requiring 10s or 1000s of individual simulation instances, each with different parameter settings correspond-

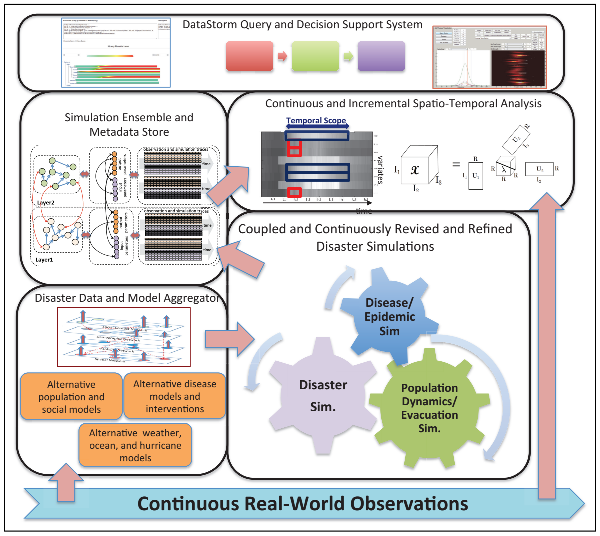

Figure 2: An overview of the `DataStorm` framework

ing to distinct plausible scenarios. Yet, while existing simulation systems provide decision support for well-specified scenarios, when decision making and knowledge discovery in the presence of incomplete information are considered, there is little support for simulation ensemble planning, optimization, and management. In particular, execution of simulation ensembles can be very costly, which leads to simulation budget constraints restricting the number of simulations one can include in an ensemble. To support effective decision making, one must answer the question "*Given a parameter space and a fixed simulation budget, which simulation instances we should the ensemble include in order to obtain models with good fit and low complexity?*" In addition, since simulation context can dynamically and unpredictably evolve over time (due to for example how a disaster develops and the preventive and reactive actions taken by individuals), continuous adaptation of simulation ensembles is necessary. In particular, as the data arrives in a streaming fashion, simulation ensembles need to be continuously revised and refined as the situation on the ground changes: (a) revisions involve incorporating real-world observations as well as updated probability densities into existing simulations to alter their outcomes; (b) refinements include identifying new simulations to run, incorporating the changing situation on the ground to support improved decision-making.

1.2 DataStorm and DataStorm-FE

In this demonstration, we introduce the `DataStorm` (Figure 2) framework for creation, storage, analysis, and exploration of coupled, multi-model simulation ensembles. At the core of `DataStorm` is the `DataStorm-FE` data- and decision-flow engine and coordination engine for creating and maintaining coupled, multi-model simulation ensembles. `DataStorm-FE` provides a means for coordinating data- and decision flows among partially-connected simulation engines, and enables end-to-end ensemble planning and optimization (including parameter-space sampling, output aggregation and alignment, continuous data streaming, and state and provenance data management) to improve the predictive accuracy of the overall end-to-end simulation process within a limited simulation budget.

Additionally, `DataStorm` provides a decision support infrastructure for results obtained by `DataStorm-FE`, permitting users to query, explore, or visualize relevant data to facilitate the decision-making process.

2. RELATED WORK

Scientific workflow systems, such as Kepler [2] and Taverna [4], support control- and data- oriented workflows that process (and integrate) large amounts of scientific data to support the scientific enterprise. The focus of these scientific workflow tools is to describe and implement data transformations and other processing needed to support scientific analysis. In general, however, these systems do not consider scenarios requiring continuous execution, nor do they provide multi-instance execution or simulation sampling. The CONFLuEnCE system [9] builds upon Kepler to add continuous execution support, but does not consider an extension to generate and execute simulation ensembles.

The use of simulation ensembles to improve predictive accuracy has been examined in the literature [7], but is usually restricted to ensembles within a homogeneous domain. The `WIFIRE` project [10] couples several heterogeneous models, but does not provide ensemble support or prioritize extensibility to other domains. The use of heterogeneous simulation ensembles for epidemics has previously been explored by [12] and [14]. However, these systems are designed around specific domain simulators; `DataStorm-FE` attempts to increase the generalizability of this approach to any potential simulation. The analysis of high-dimensional time series data in the context of heterogeneous ensembles has also been explored by [15], with emphasis on tensor-based approaches, but generation of ensembles is not addressed.

3. SYSTEM ARCHITECTURE

In `DataStorm-FE`, the inputs to a decision-flow are the data sources, with each step in the decision-flow involving an analytic function, model, or decision criterion. Inputs to each step are data and decisions from the previous steps, plus user-provided decision parameters. The resulting data and decisions are outputs. The sinks of the flow are the simulation instances and alternative conclusions based on the data and decision parameters supplied by the user.

3.1 Overview

At its lowest level, `DataStorm-FE` extends the Kepler scientific workflow system [2], designed to integrate disparate models into a unified whole; in particular, as in Kepler, *actors* provide code modules that execute a particular task, with the data flow between these actors being controlled by a single global *director*. While Kepler provides a flexible framework to create executable scientific workflows, including an actor-oriented modeling paradigm, tools for data transformation and access, and a GUI for the design of scientific workflows, it has significant limitations: (a) Kepler is not designed for ensemble executions; it can only take a fully-instantiated model and execute a specified workflow; (b) Kepler does not provide stateful actors, and is not designed for continuous workflow executions; and (c) while Kepler's Web and Grid service actors allow scientists to utilize computational resources on the net in a distributed scientific workflow, it does not provide native support for parameter space sampling, distributed instantiation, and parallel execution of simulation instances in an ensemble.

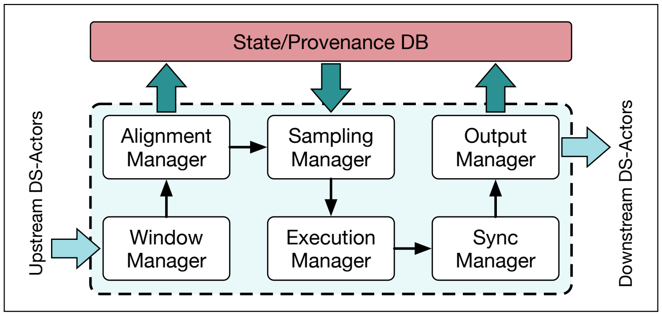

Figure 3: A `DataStorm-FE` actor (DS-actor), with individual sub-modules.

3.2 DS-Actors and DS-Flows

The data- and decision-flows (i.e., DS-flows) supported by `DataStorm-FE` are inherently temporal. Therefore, rather than requiring a single-shot execution, where inputs for the workflow are consumed and actors are invoked only once, DS-flows benefit from continuous execution, where data is consumed continuously and each actor is invoked repeatedly, as up-stream actors produce new results for analysis.

The core component of the `DataStorm-FE` system is the `DataStorm`-Actor or DS-actor, consisting of several more specialized sub-modules which adapt the domain simulator to interface with the wider system (Figure 3). Each DS-actor is *stateful*, in the sense that, the actor records its own state variables and outputs, and is able to recall its previous state before each execution cycle. As data (represented as multi-variate time series) flows through the system, it is encoded as structured BSON including data and metadata, the provenance of which is automatically recorded by the system, for later use and analysis.

These results may be generated individually by a single model instance, or in aggregate by the ensemble collectively. To differentiate these types, we refer to the former as DS-Individual Results (DS-IR), and to the latter as DS-Aggregated Results (DS-AR). DS-IRs contain references to raw output data, spatial and temporal context for that data, and other metadata required for replication. A DS-AR is a collection of one or more DS-IRs, with a spatiotemporal context equal to the union of its constituent DSIRs. Note that, if real-world conditions change and new observations or inputs are produced, a scenario may need to be re-simulated. In an ensemble, however, any changes will cascade through the system, potentially requiring a re-computation of every downstream simulation. Data provenance also helps identify which subsets of data are affected by a given change, thereby helping reduce redundant work: simulation plans can be created to re-run only the affected data, reducing execution time. Stateful modules may leverage these advantages further, by adjusting parameters mid-execution or aborting redundant executions.

3.2.1 Data Pre-Processing

Figure 3 presents the internal structure of a DS-actor. The relevant data from upstream DS-actors and external sources are ingested and aligned before used for simulations. The *WindowManager* handles temporal window alignment between models, ensuring sufficient data exist for downstream execution, repackaging upstream results into temporally-aligned subsets. The *AlignmentManager* receives these subsets, converts the raw results into a format the running model can process, and handles spatial realignment. It then stores these transformed results in the provenance database for later use during sampling.

3.2.2 Ensemble Creation

In the next step, the *SamplingManager* samples the possible simulation space, balancing computation budget against potential scenario fan-out, to decide which simulation scenarios to execute. As uncertainty within coupled simulations will result in an exponential increase in possibilities, these must be pruned prior to execution to prevent an unscalable explosion of simulated scenarios. Additionally, the simulation budget may take into account metadata associated with the computational costs of each model, providing a wider fan-out for cheaper simulations and more aggressive pruning for expensive ones.

Once the simulation scenarios to be executed have been identified, the *ExecutionManager* executes these configurations within the cluster, allocating and load balancing instances as necessary. Each scenario selected by the sampler for simulation must be instantiated and modeled on a cloud-based instance, in order to preserve the per-model scalability and extensibility of the system, and to permit parallel processing of many different simulations simultaneously. However, cloud resources generally carry high operational costs. Therefore, `DataStorm-FE` provides an adaptive orchestration layer (using Ansible to automate configuration management, and Vagrant for provisioning and deployment) capable of creating, configuring, and destroying instances as needed to balance performance and budget requirements.

3.2.3 Data Post-processing

Once the simulation instances have been executed, the resulting data has to be collected, aggregated, and transformed before being passed to the downstream DS-actors. The *SyncManager* permits users an opportunity to synchronize, aggregate, and transform the outputs of intrinsically-asynchronous models. Finally, the *OutputManager* receives the results from the model, then packages and stores the results in the provenance database before passing control to the next downstream DS-actor.

4. DEMONSTRATION SCENARIO

In this demonstration, we use a hurricane-based disaster as a sample scenario for `DataStorm-FE`. Disasters pose significant challenges for emergency planning and management as effective disaster response requires matching available resources to shifting demands on a number of fronts. The recent hurricanes in the US highlight the importance of predictive and real-time response and decision making. Effectively managing current and future emergencies through real-time and continuous decision making requires data- and model-driven computer simulations for predicting the evolution of disasters and related hazards. However, data uncertainty, interaction complexity, and resource constraints have thus far proved to be significant roadblocks to widespread adoption of these techniques. Simulation models frequently predict only a few results, without regard for the 1000s of interdependent variables in an emergent disaster area, while specialized domain knowledge requirements complicate the development of integrated simulators. The sheer quantity of simulation results, coupled with real-world time and computational constraints, pose further challenges.

4.1 Simulated Models

The three disaster-related simulation models selected cover hurricanes, flooding, and human mobility. These models were selected for both their relative frequency and their semi-dependent nature. With the Weather Research and Forecasting (WRF) hurricane model [6], we are able to predict the track of a hurricane, as well as associated wind speeds and rainfall. Since this model does not attempt to predict flooding related to these events, we couple the output with Itzï [13], a hydrological simulator which models the flow of flood-waters over the landscape. Additionally, the hurricane and floods have behavioral impacts on the affected population; therefore, the corresponding DS-actors are coupled with the Opportunistic Network Environment simulator [5], which is used to track population movement through transport networks within the affected area.

4.2 Data Flow

For our demonstrated simulation, the user selects several possible hurricane scenarios, which correspond to distinct inputs to the `hurricane` DS-actor. Multiple simulator instances spin up and evaluate the proposed scenarios, producing different rainfall maps, wind speeds, and hurricane tracks, for a given spatio-temporal frame. These outputs are aggregated and passed to the downstream actors. These are spatiotemporally aligned, then converted to match the expected inputs to the flooding and mobility models. For the `flood` DS-actor, the sampler chooses a subset of the flood scenarios to be executed and these scenarios are pushed to the executor to generate the corresponding flood simulation instances. The `mobility` DS-actor waits until the outputs from both `hurricane` and `flood` DS-actors are ready. Once the data are pre-processed, the sampling, alignment, and conversion processes repeat again for the `mobility` DS-actor to generate the corresponding simulation ensemble. In the continuous execution, this process continues with DS-actors producing new simulation ensembles based on new or revised data from upstream DS-actors.

4.3 Visualization

At any time, the visualization tool may be used to view the currently-available results from the provenance database, if any. Spatial adjustments, such as zooming or panning, can be accomplished on the map; temporal adjustments are available through the timeline scrubber. Additionally, different simulations may produce variable outputs, which manifest as distinct visualization layers which can be grouped or hidden. This permits domain experts to narrow their evaluation to relevant data, simplifying analysis, while still permitting administrative staff a higher-level strategic evaluation of the situation as it evolves.

5. CONCLUSIONS

Simulation-based decision making requires the ability to acquire, integrate, model, analyze, index, and search, in a scalable manner, large volumes of multi-variate, multi-layer, multi-resolution, interconnected, and inter-dependent spatio-temporal data produced by simulations. In this demonstration, we introduce the `DataStorm` framework for coupled, multi-model simulation ensembles and the `DataStorm-FE` data- and decision-flow and coordination engine for creating and maintaining coupled, multi-model simulation ensembles.

Acknowledgements

We thank Lu Cheng, Ketut Gita, Shengyu Huang, Huan Liu, David Maitha, Logan Mathesen, Pitu Mirchandani, Giulia Pedrielli, Jason Truong, Dalton Turner, and Ming Zhao of ASU for their contributions. Results presented in this paper were partially obtained using the Chameleon testbed supported by the National Science Foundation.

6. ADDITIONAL AUTHORS

Xinsheng Li (ASU, `xinshen@asu.edu`), Sicong Liu (ASU, `sliu104@asu.edu`), Nicholas Martinez (ASU, `nlmarti4@asu.edu`), Jiayong Mo (ASU, `jiayongm@asu.edu`) Elliot Nester (ASU, `enester@asu.edu`), Silvestro Poccia (UNITO, `poccia@di.unito.it`), Manjusha Ravindranath (ASU, `mravind1@asu.edu`), and Maria Luisa Sapino (UNITO, `mlsapino@di.unito.it`).

7. REFERENCES

[1] Committee on environment and natural resources, grand challenges for disaster reduction, natl sci. and tech. council, executive office of the president, 2008.

[2] Altintas, I *et al.* Kepler: an extensible system for design and execution of scientific workflows. In *SSDM*, 2004.

[3] Ford, DA *et al.* An extensible spatial and temporal epidemiological modelling system. *International Journal of Health Geographics*, 2006.

[4] Hull, D *et al.* Taverna: a tool for building and running workflows of services. *Nucleic acids research*, 2006.

[5] Keranen, A. Opportunistic network environment simulator. *Special Assignment Report, Helsinki Univ. of Tech.*, 2008.

[6] Davis, C *et al.* Prediction of landfalling hurricanes with the advanced hurricane wrf model. *Monthly weather review*, (6), 2008.

[7] Olsen, KB *et al.* ShakeOut-D: ground motion estimates using an ensemble of large earthquakes on the southern san andreas fault with spontaneous rupture propagation. *Geophysical Research Letters*, 2009.

[8] Balcan, D *et al.* Modeling the spatial spread of infectious diseases: The global epidemic and mobility computational model. *Journal of Comp. Sci.*, 2010.

[9] Neophytou, P *et al.* Confluence: Continuous workflow execution engine. In *SIGMOD*, 2011.

[10] Cowart, C *et al.* geokepler workflow module for computationally scalable and reproducible geoprocessing and modeling. In *AGU Fall Meeting Abstracts*, 2015.

[11] Liu, S *et al.* NOTES2: networks-of-traces for epidemic spread simulations. In *AAAI*, 2015.

[12] Liu, S *et al.* epiDMS: data management and analytics for decision-making from epidemic spread simulation ensembles. *The Journal of infectious diseases*, 2016.

[13] Courty, LG *et al.* Itzï (version 17.1): an open-source, distributed gis model for dynamic flood simulation. *Geoscientific Model Development*, 2017.

[14] Poccia, S *et al.* SimDMS: data management and analysis to support decision making through large simulation ensembles. In *EDBT*, 2017.

[15] Li, X and Candan, KS and Sapino, ML. M2TD: multi-task tensor decomposition for sparse ensemble simulations. In *ICDE*, 2018.