Proceedings of the 2020 Winter Simulation Conference
K.-H. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, and R. Thiesing, eds.

SYSTEM INTEGRATION WITH MULTISCALE NETWORKS (SIMON): A MODULAR FRAMEWORK FOR RESOURCE MANAGEMENT MODELS

Marisa Hughes
Michael Kelbaugh
Victoria Campbell
Elizabeth Reilly
Susama Agarwala
Miller Wilt
Andrew Badger
Evan Fuller

The Johns Hopkins Applied Physics Laboratory
11100 Johns Hopkins Rd
Laurel, MD 20723, USA

Ximena Calderon Arevalo
Alex Fiallos
Lydia Fozo
Jalen Jones
Dillon Ponzo

The Johns Hopkins University
3400 N. Charles St
Baltimore, MD 21218, USA

ABSTRACT

Although the scientific community has proposed numerous models of Earth and human systems, there are few tools available that support the model coupling that is necessary to capture their complex interrelationships and promote further research cooperation. To address this challenge, we propose System Integration with Multiscale Networks (SIMoN), an open source modeling framework with a novel methodology for supporting heterogeneous geospatial regions. SIMoN enables users to define consistent aggregation and disaggregation maps for transformation between disparate notions of geospatial units such as counties, watersheds, and power regions. We have applied this unique tool to couple models across domains including as climate, population, and food-energy-water (FEW) systems.

1 INTRODUCTION

With the rise of globalization, climate change, population growth, and resource depletion, new modeling techniques are needed to assess the sustainability of our future resources. These methods must adapt to highly coupled domains, accommodate new models and data as they emerge, facilitate validation through model comparison, and handle the patchwork of data and models available with different units, definitions, and geo-temporal scales.

Currently, many researchers focus on single domain models at incompatible geospatial scales, which poses a challenge in predicting the far-reaching impacts from changes or failures in a single system. We introduce a new framework – System Integration with Multiscale Networks (SIMoN) – for joint model runs that addresses the challenge of data transformation between models with disparate geospatial definitions. The SIMoN framework assists modelers in predicting the availability of critical resources such as power, water, and food in the future. The framework also incorporates population and climate change models, as these are strong drivers for resource demand and availability. These domains each come with their own hierarchies of geospatial regions which include political, topographical, regulatory, and coordinate grid induced boundaries.

Here, we outline our approach to combine these many definitions into a general notion of geospatial partitions that organize the problem. We also demonstrate the capability of SIMoN to integrate models and data from disparate domains by predicting water availability in 2050, as it depends on population growth,

climate change, and corresponding increases in demand for thermoelectric cooling. Multiple modeling architectures are shown which have been run in SIMoN, demonstrating its flexibility for model exchange.

In addition to our modeling accomplishments, we will describe the more general mathematical and software framework used to integrate data and models that operate on heterogeneous geospatial regions. We define types of transformation functions, called aggregators and disaggregators, to exchange data across previously incompatible systems. These aggregators and disaggregators must conform to a set of axioms, designed to reduce the propagation of error and provide a provable notion of consistency. Such functions are reusable for connecting different model sets into a joint modeling architecture.

The SIMoN framework is designed to be extensible and flexible, including tools to enable the introduction of new domains. We expect the set of geospatial definitions, tranformation functions, and domain models available in SIMoN to grow as it is applied to further challenges in cross-domain integration.

2 PREVIOUS WORK

Integrated approaches to modeling the food-energy-water (FEW) nexus have been increasing for the past decade or two. For instance, Lubega and Farid created a physics-based model for the water-energy nexus using the Systems Modeling Language (SysML) (Lubega and Farid 2014). Additionally, Tidwell et al created a system dynamics model for integrated management and decision support of electrical and water systems in the US (Tidwell, Kobos, Malczynski, Hart, and Klise 2009). Some efforts even brought together siloed decision support modeling systems for a more integrated approach. For example, Yates and Miller (Yates and Miller 2013) linked the Water Evaluation and Planning (WEAP) modeling system (Lee, Sieber, and Swartz 2005) with the Long Range Energy Alternatives Planning (LEAP) system for integrated energy and water planning. Most recently, Endo et al (Endo, Tsurita, Burnett, and Orencio 2017) provided an overview of research in the area of food, energy, and water, including tools developed for modeling and analysis.

Current approaches to modeling FEW and related systems are generally tailored to the subset of systems that they address and are not designed for adaptation to new or different models within these sectors. As such, they also do not address between-model differences in geospatial or temporal scales that may occur when marrying two or more models that were not initially designed to work together or that were developed in isolation. Further, due to the complicated nature of the interactions between systems, as well as the enormous extent of each individual system, many researchers focus on two of these systems at a time (Bazilian, Rogner, Howells, Hermann, Arent, Gielen, Steduto, Mueller, Komor, Tol, et al. 2011). Additionally, current approaches do not aim at developing a flexible framework that would allow further systems to be included in analysis, but rather focus on development of individual models and their interactions.

There are more general modeling and simulation tools used in this space, though none filling all the gaps mentioned in the previous paragraph. For instance, consider Dymola (Dempsey 2006), a component modeling tool that models physical parts and thus operates exclusively at a physical level. The General Algebraic Modeling System (GAMS) (Bussieck and Meeraus 2004) is a powerful optimization tool that does not aid modellers in handling different temporal and geospatial scales. The Climate, Land, Energy, and Water (CLEW) modeling framework (Hermann, Rogner, Howells, Young, Fischer, and Welsch 2011) is concerned with resource modeling in order to identify relationships between sectors and how to minimize trade-offs. It may be used at different geospatial scales, though models in each sector must agree on a single geospatial scale. The Multi-Scale Integrated Analysis of Societal and Ecosystem Metabolism (MuSIASEM) (Giampietro, Mayumi, and Ramos-Martin 2009) allows for modeling at multiple scales with a focus on the performance of socioeconomic activities and ecological constraints in order to understand and societal analyze resource use its impacts on the environment. The Global Change Assessment Model (GCAM), developed as a collaboration between the University of Maryland and the Pacific Northwest National Laboratory (PNNL), is another integrated assessment tool for exploring the impacts of global change (Edmonds and Reiley 1985; Edmonds, Wise, Pitcher, Richels, Wigley, and Maccracken 1997). The

individual GCAM models, as well as the temporal and geospatial scales are fixed within the larger GCAM model. Furthermore, simultaneous solvers couple the GCAM domains tightly. The approach laid out in this paper allows the user to more flexibly combine models without simultaneously solving over the same geospatial regions.

3 MODEL ASSUMPTIONS AND INITIAL MODELS

SIMoN models are assumed to work together to predict the supply, demand, or distribution of resources in the future using discrete time steps. The goal of SIMoN is to provide tools for combining existing models. Thus, initial SIMoN experiments have used publicly available data sets and relatively simple Python models for proof-of-concept. The SIMoN framework is open source and includes a domain model library with example models for various domains including power, water, and population. The framework captures the most important interactions between these models in the interface described in Section 8: Software Framework. Models currently implemented for demonstration include:

- The **population model** uses Holt's linear regression method (Hyndman and Athanasopoulos 2018) (implemented in the statsmodel Python package) to predict the population of each county. It extrapolates US Census Bureau population data (United States Census Bureau 2018) from 2000 to 2016 into the future, making a population prediction for each future year. The model gives more weight to the most recent historical data, so the population change from 2015 to 2016 is more significant than the change between 2000 and 2001.
- The **power demand model** calculates each county's power demand by multiplying its population by its state's power consumption per capita based on total state-level power sales (U.S. Energy Information Administration 2019a).
- The **power supply model** calculates the carbon dioxide emissions and thermoelectric water usage of each North American Electric Reliability Corporation (NERC) sub-region's power production by assuming that power supply can freely shift to meet power demand in equilibrium (supply = demand, production = consumption) at a constant price. The provided energy profile, generated using U.S. EIA plant-level emissions and water consumption data (U.S. Energy Information Administration 2019b), gives each NERC sub-region's pre-calculated rates of emissions and water usage per MWh of energy production. Each NERC sub-region's power demand is multiplied by its profile rates to determine its total carbon dioxide emissions and its thermoelectric water usage, for the level of power that is demanded and produced.
- The **water demand model** calculates each county's water demand by multiplying its population by its water consumption per capita, and then adding the county's thermoelectric water usage from the power supply model's output. Water consumption rates for each county were calculated by subtracting "thermoelectric recirculating, total consumptive use, fresh in Mgal/d" from "irrigation and thermoelectric water, total consumptive use, fresh in Mgal/d". The difference was divided by the county's 2015 population, then multiplied by 365 to convert the daily rate to the annual rate.
- The GFDL CM3 **climate model** (Griffies, Winton, Donner, Horowitz, Downes, Farneti, Gnanadesikan, Hurlin, Lee, Liang, et al. 2011), published by the National Oceanic and Atmospheric Administration (NOAA), uses representative concentration pathways to determine atmospheric conditions and its effects on temperature, precipitation, and evaporation. The SIMoN model does not perform any of these actual calculations, but simply retrieves pre-calculated data from the config file.

Another major model under development includes a **water supply model**, which follows the general guidelines that are used to determine water resource budgets for each area as described by the Michigan Water Resources Department assuming a steady state water budget (Michigan Water Resources Division 2010). Water availability is derived from input (rainfall) and consumption (evaporation, evapotranspiration, and

human use) for each Hydrologic Unit Code 8 (HUC8) watershed, where both consumption and geospatial definitions are found in public USGS data sets (Unites States Geographical Survey (USGS) 2019). An alternative **population model** uses a logistic regression approach. The FAIR **climate model** (Smith, Forster, Allen, Leach, Millar, Passerello, and Regayre 2018) can act as an alternative to the GFDL CM3 climate model.

It is up to the modeler running the SIMoN framework to decide which models to use and how these models should interact within the framework. The examples of data exchange given here can be enriched or reduced. A strength of SIMoN is the ability of a modeler to easily test several simulation scenarios by swapping out different models while maintaining compatibility across the system. Figure 1 demonstrates two configurations of SIMoN for the contiguous United States using different climate models. One uses Climate RCP predictions while the other uses the FAIR climate model. When a new model is included, the modeler must redefine the interactions of the new model with the rest of the system so that inputs and outputs are shared appropriately across models.

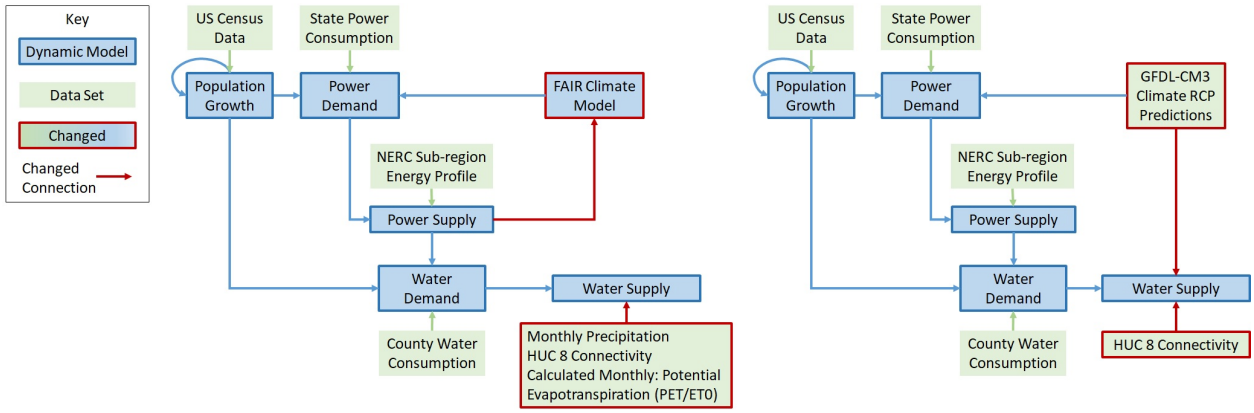

Figure 1: Left, SIMoN uses the FAIR climate model which incorporates CO2 emissions from the power sector. On the right, SIMoN uses precomputed outputs of GFDL CM3. The arrows show data flow.

4 GEOSPATIAL DEFINITIONS

The SIMoN framework is designed to capture the interactions between domains such as power, water, population, climate, and food. Each of the domains modeled in SIMoN has its own definitions of geospatial regions. In fact, domains tend to have entire hierarchies of geospatial partitions associated to them, as this is a natural way to organize data collection and curation. For example, the U.S. Census Bureau organizes its data collection into a complex hierarchy: https://www2.census.gov/geo/pdfs/reference/geodiagram.pdf?#. Census definitions include nested definitions such as census tracts, counties, and states, but also include unusual and potentially overlapping regions such as voting districts, and urban growth areas.

While we wish to be flexible to many definitions of geospatial regions, we must establish some rules for allowable geospatial definitions to enable data transformations. It is required that all models in a given simulation using the SIMoN framework operate on the same overall geospatial region called the **scope** (e.g. the United States). Any data outputs shared by a model for the consumption of other models in SIMoN must include data for the entire scope. Each model, however, may have different definitions of how that scope is subdivided. In particular, any data set can be shared on any finite partitions of the scope. Elements of the partition are non-overlapping subsets of the scope, where the union of these elements is the entire scope. Examples of partitions include counties, watersheds, NERC or other power regions, lat/lon grid overlays, map pixels, etc. For the purpose of initial demonstration, the examples that follow will use the contiguous United States as their geospatial scope. Many of the region types captured in the Census Bureau Hierarchy are examples of partitions of this scope. However, Urban Areas are NOT a partition

of the United States, since not every point in the United States is contained in an Urban Area. A model concerned with Urban Areas would have to be extended (even if that extension is trivial) in order to operate with other models over the contiguous United States within SIMoN.

Given multiple partitions of the overall geospatial region, some will be naturally compatible in that they represent refinements of others. Given two partitions $X = \{x_i\}_{i=1}^{n}$ and $Y = \{y_j\}_{j=1}^{m}$ of the same scope S, we say that Y is a refinement of X if $\forall y_j \in Y, \exists x_i \in X$ such that $y_j \subseteq x_i$. For example, counties are a refinement of states, because both partition the contiguous United States and each county is contained in a single state. Note: There exists two special cases of geospatial definitions that are modified to satisfy the partition rules - (1) Louisiana uses parishes instead of counties to partition the state and in SIMoN they are in the same partition definition of "counties", and (2) the District of Columbia is considered neither a state nor county by formal definitions and exists in SIMoN on both "state" and "county" geospatial partitions without further refinement.

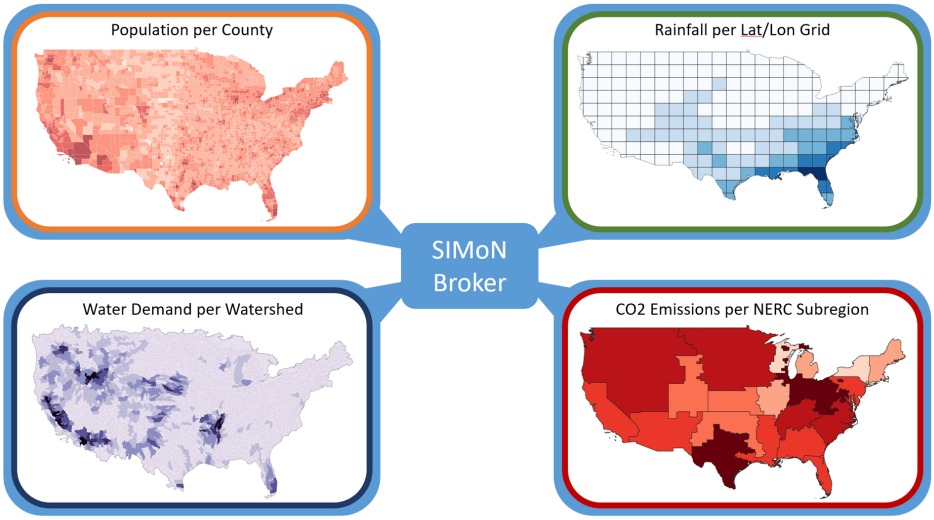

Figure 2: An example SIMoN output from four integrated models predicting the year 2050. SIMoN performs aggregation and disaggregation functions to convert data between geospatial definitions when data are passed between models. The SIMoN Broker is detailed in Section 8.

Figure 2 demonstrates the important role that SIMoN can play when modeling the Earth's systems. In this example, there are four models and datasets: population projection, rainfall data, water demand, and power supply. Each model operates according to its own geospatial definition (county, lat/lon grid, watershed, NERC subregion, respectively). Yet, in the overall system, these models need to share data. SIMoN acts as a broker between the models, performing necessary transformation to allow model communication.

5 ABSTRACT GEOSPATIAL CONSTRUCTION

Given the heierarchical nature of the data, we construct a sequence of directed graphs of geospatial regions to create a reference tool for models to communicate through the framework. In particular, a directed graph has unidirectional arrows, from a geospatial region to another that contains it. These graphs and functions defined on their edges organize the data transformations between model geospatial regions and are used to define a global notion of consistency of data as it is transformed between heierarchical levels.

Any set of partitions of the scope S forms a partially ordered set, or poset, where the binary relation $\leq$ is given by $Y \leq X$ if and only if Y is a refinement of X. It is straightforward to check that this operation is reflexive ($X \leq X$), antisymmetric ($X \leq Y; Y \neq X \Rightarrow X \not\leq Y$) , and transitive ($X \leq Y \leq Z \Rightarrow X \leq Z$). This partially ordered set can be represented uniquely by its Hasse Diagram: a directed acyclic graph (DAG),

or a directed graph with no paths starting and ending at the same vertex that respects the directionality of the edges, where the vertices represent partitions and an edge $Y \to X$ is inserted if Y is a refinement of X and there are no intermediate partitions (a transitive reduction). We select the convention of always drawing the finest partitions at the bottom, and the scope (supremum) at the top.

In general, we will define P to be the partially ordered set of the scope and its partitions of interest, that is- those partitions of geospatial regions which represent possible data granularities in SIMoN. An example P is given in Figure 3, represented by the corresponding Hasse Diagram in the form of a directed acyclic graph G_P.

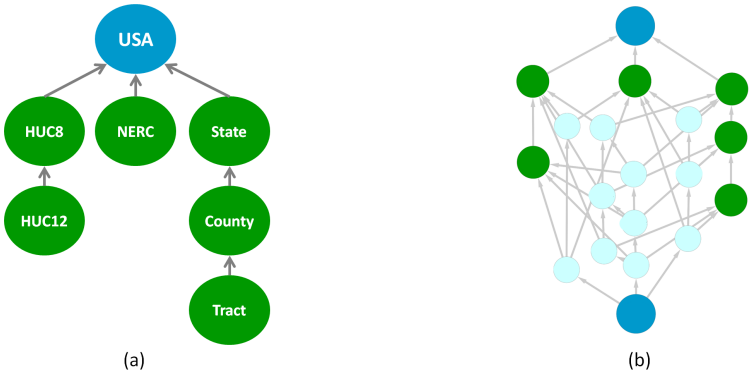

Figure 3: (a) An example Hasse Diagram G_P with water, power, and political regions represented from left to right. (b) The corresponding extended diagram G_Q with intersection regions and an infimum added.

Given a P representing the scope S and its geospatial definitions of interest from the data/model perspective, we construct an augmented poset, Q, which includes the meets of all pairs geospatial definitions in P. An example of such meets in a Q is seen in the light blue vertices in Figure (3)b. Specifically, for $X, Y \in P$, $X \wedge Y$ is the largest element in Q such that $X \wedge Y \leq X$ and $X \wedge Y \leq X$. If X and Y are incomparable in P, i.e. whenever $X \not\leq Y$ and $Y \not\leq X$, $X \wedge Y \in Q \setminus P$. However, if $Y \leq X$ in the poset P, $X \wedge Y = Y$, i.e. a new element is not introduced to the poset. We also add a universal meet, or infimum, called $INF \in Q$ to P. By construction, the operation of meet is symmetric so $\forall X, Y \in P, X \wedge Y = Y \wedge X$. Including the universal meet, we add at most $\binom{n}{2} + 1$ elements to $|P|$ to construct Q. If $|P| = n$, then $|Q| \leq n + \binom{n}{2} + 1 = \binom{n+1}{2} + 1$.

The purpose of constructing Q is to define a new geospatial partitions which serve as a bridges between their parent elements. Each $X \wedge Y \notin P$ represents a new, unique geospatial partition of the scope which is the maximal partition that is a refinement of both X and Y. To transform data from geospatial granularity X to granularity Y, it is necessary to follow the undirected path $(X, Y \wedge X)(X \wedge Y, Y)$, using operations defined in Section 7. The universal meet, while not directly involved in any data transforms, will be useful in checking the consistency of geospatial regions and function definitions. Visually, if the boundary lines of X and Y are overlaid on the same map, all bounded regions are elements of the partition $X \wedge Y$. Because X and Y are partitions, for each region $z \in X \wedge Y$, $\exists$ unique $x_i \in X, y_j \in Y$ such that $z = x_i \cap y_j$. We call x_i and y_j the parents of z, just as we call X and Y the parents of $X \wedge Y$. We associate to Q an abstract DAG, G_Q representing its Hasse diagram. The graph G_Q is our final definition of the relationships between geospatial definitions, and is called the **Abstract Geospatial Graph**, as shown in Figure 3b.

The advantage of constructing these pairwise and universal meets will be that they provide bridges for transforming data between the geospatial definitions of different models. A motivating example is provided in Figure 4.

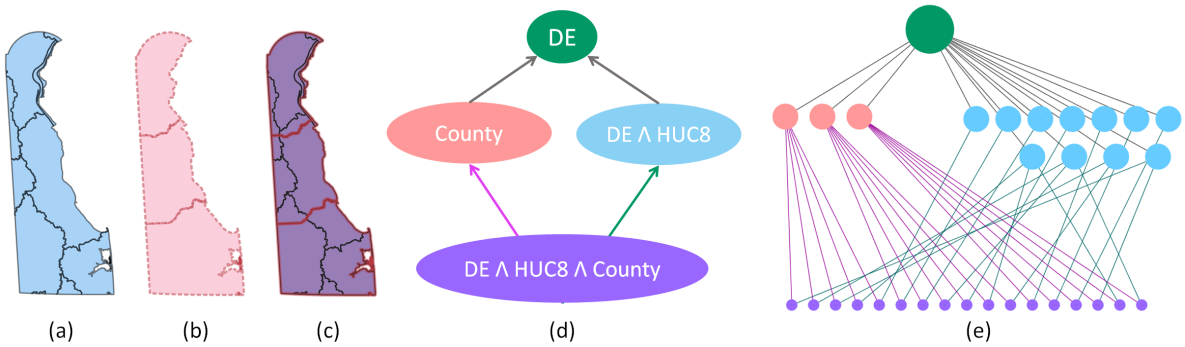

Figure 4: Here we see three geospatial definitions of the state of Delaware(DE). On the left, maps of DE include (a) 11 HUC8 Regions instersected with DE, (b) 3 DE Counties, and (c) the 17 intersection regions DE $\wedge$ HUC8 $\wedge$ County, where both sets of lines are considered to bound regions. The corresponding graphs capturing the relationships between these geospatial regions are shown on the right, where (d) is the Abstract Graph with a scope of Delaware, and (e) is the Instance Graph with edges representing inclusion.

6 CONSTRUCTING GEOSPATIAL DATA AND INSTANCE GRAPHS

The directed graphs representing abstract definitions of geospatial regions and their relationships in the previous section have real world instantiations of interest to modelers. We will assume that $\forall X \in P$ representing a model-driven definition of geospatial regions, there is a shapefile SF_X that fully defines the regions of X within the scope S. For example, if X is states in the contiguous United States, then SF_X will be a shapefile with 49 distinct regions. We represent each of these regions as a node in the **Instance Geospatial Graph** I_P. The instance graph has the same general structure as the abstract geospatial graph, except that every individual vertex in the abstract graph has been replaced by many vertices as in Figure 4e. When considering our larger example of the continental United States, rather than a single vertex representing 'states' in our political boundary graph, there would be 49 vertices of type 'state' representing the forty-eight contiguous states and the District of Columbia. Each of these states has a directed edge into the single element of the scope/supremum. Similarly, the node 'county' is replaced with 3108 vertices of type 'county', each with an edge to its unique parent of type 'state'.

To construct the instance graph I_P corresponding to P, a shapefile is provided for every vertex $X \in P$ in each of the shapefiles that correspond to the pairwise meets $X \wedge Y$ where $X, Y \in P$ can be accomplished more efficiently by iterating from top to bottom, and using the graph structure to reduce the intersections computed. Instance graphs grow large quickly, so computing them up front rather than at run-time can provide a significant savings to the modeler. For example, the instance graph for a partially ordered set containing only the elements Nation, State, County, and NERC region consists of 7502 nodes and 15727 edges. Example instance graphs and corresponding shapefiles are provided as a service for geospatial reconciliation on the SIMoN github.

7 AGGREGATION AND DISAGGREGATION DEFINITIONS

Let Y be an vertex of the abstract geospatial graph G_Q. Let $\vec{v}_Y$ be a vector of length $|Y|$ representing a data value defined for all vertices of type Y in the instance graph I_Q. For example, $\vec{v}_Y$ may represent the population of each county where Y is the vertex 'county', or the rainfall in each watershed region where Y represents the vertex 'HUC8'.

We define an **aggregation operator** A to be a collection of functions, one for each directed edge (V,W) in G_Q that take data defined on the instances of W to data defined on the instances of V. Write the function associated to the (V,W) as $A_{\{W:V\}}$, where $A_{\{W:V\}} : \mathbb{R}^{|W|} \to \mathbb{R}^{|V|}$. These functions aggregate data from a refinement, such as counties, to a higher level, such as states. We will not assume that these functions

can be represented by matrix multiplication, although this is the case for many important examples. The most common aggregation operator will sum the data over the children of a node in the instance graph I_Q. For example, the population of a state for the power demand model will be calculated by summing the populations of its child counties which are output by the population model. However, not all aggregations are sums. For example, to aggregate the maximum power load over a set of child nodes in the instance graph, the aggregation operator MAX would be used.

Aggregation Notation: Let $Y \leq X$ in Q. If $\vec{v_Y} \in \mathbb{R}^{|Y|}$ is the data associated to Y, then $\vec{v}_X = A_{\{Y:X\}}(\vec{v}_Y)$ is the data associated to X by aggregation operator A.

Similarly, a **disaggregation operator** D is a collection of functions, $D_{\{V:W\}}$, on edge (V,W) that takes data on V to data on W: $D_{\{V:W\}} : \mathbb{R}^{|V|} \to \mathbb{R}^{|W|}$. These functions disaggregate from a node to a refinement. The following are common examples of disaggregation functions on the instance graph I_Q:

- The constant map which populates each instance child node with a copy of the data from its parent
- Subdividing a data quantity among children according to a known ratio, such as the area of each child region

Disaggregation Notation: Let $Z \leq Y$ in Q. If $\vec{v_Y} \in \mathbb{R}^{|Y|}$ is the data associated to Y, then $\vec{v}_Z = D_{\{Y:Z\}}(\vec{v}_Y)$ is the data associated to Z by disaggregation operator D.

We say that the operators (A,D) form a valid **aggregation-disaggregation pair** on G_Q if and only if they satisfy the following axioms:

1. **Consistency**: For $Y \leq X$ in Q, and any $x_i = y_j$, the aggregators and disaggregators are the identity on the j^{th} and i^{th} components respectively:
   $\forall \vec{v}_Y \in \mathbb{R}^{|Y|}, \quad \vec{v}_X[i] := A_{\{Y:X\}}(\vec{v}_Y)[j] = \vec{v}_Y[j]$
   $\forall \vec{v}_X \in \mathbb{R}^{|X|}, \quad \vec{v}_Y[j] := D_{\{X:Y\}}(\vec{v}_X)[i] = \vec{v}_X[i]$
2. **Left Inverse**: $\forall Z \leq Y, A_{\{Z:Y\}} \circ D_{\{Y:Z\}} = \mathrm{id}_{\mathbb{R}^{|Y|}}$.
3. **Transitivity**: $\forall X \leq Y \leq Z \in Q, D_{\{X:Z\}} = D_{\{Y:Z\}} \circ D_{\{X:Y\}}$

The example operators pairs, (MAX, constant) and (SUM, area weighted subdivision) form valid (A,D) pairs. We show this explicitly for the pair (MAX, const). To check consitency, note that if $x_i = y_j$, MAX sets $\vec{v}_X[i] = \max_{y_j \subseteq x_i}\{\vec{v}_Y[j]\}$. Since there is only one such y_j, $\vec{v}_Y[j] = \vec{v}_X[i]$. Simlalrly the constant maps sets $\vec{v}_Y[j] = \vec{v}_Y[j]$. To check left inverse, the constant function assigns a particular value, $\vec{v}_X[i]$ to all $y_j \subseteq x_i$. Therefore, when the MAX is taken over all $y_j \subseteq x_i$, $\vec{v}_X[i]$ is again assigned to x_i. To see transitivity, note that the constant map assigs the value $\vec{v}_X[i]$ first to all $y_j \subseteq x_i$ and then to all $z_{k,j} \subseteq y_j$. This is equivalent to assigning the value $\vec{v}_X[i]$ to all $z_{k,j} \subset y_j$. The arguments to see that (SUM, area weighted subdivision) is a valid (A,D) pair are simlar.

These definitions provide a coherent mechanism for transferring a data vector between any two partitions. Suppose we are given partitions $X,Y \in Q$ and a data value $\vec{v}_X \in \mathbb{R}^{|X|}$ that we want the system to publish for partition Y. In general, we extend the data $\vec{v}_X$ to all of Q using a valid aggregator and disaggregator pair, (A,D). By construction, G_Q also contains $X \wedge Y$. We obtain $\vec{v}_{X\wedge Y} = D_{\{X:X\wedge Y\}}(\vec{v}_X)$, then apply $A_{X\wedge Y:Y}$ get the data vector $\vec{v}_Y \in Y$. That is, $\vec{v}_Y = A_{\{X\wedge Y:Y\}} \circ D_{\{X:X\wedge Y\}}(\vec{v}_X)$. The axioms ensure consistency on any region z that is part of X and Y. In particular, if $X = Y$, then we obtain the original data vector.

A different aggregation-disaggregation pair may be defined by the modeler for each data variable exchanged over the SIMoN framework. These mappings will be used by the outer wrappers that surround each model to transform data between geospatial definitions as needed. As more modelers implement valid transformation pairs, the library of recommended pairs will grow. We further plan to implement tools to check that potential new operator pairs satisfy the axioms listed here. Algorithmic approaches to the efficient construction of the data structures and consistency checks proposed are currently under development.

Returning to the example of regions in Delaware, shown in Figure 4, we can define a number of aggregation-disaggregation pairs for illustration. For example, if Kent, Sussex, and New Castle counties have populations of 160k, 200k, and 540k people respectively, then the total population of the state can be aggregated simply by summing the county populations, resulting in a state total of 900k. Likewise, if we begin with a state population of 900k, we can estimate county populations using multiple disaggregation methods. One method would be to weight each of the 3 counties equally, and assign each county one-third of the state population, or 300k people. Alternatively, we might expect that a county's population is roughly proportional to its area, and assign each county a proportion of the state population that is equal to its share of the state area. The resulting distribution would be approximately 266k, 417k, and 217k people for Kent, Sussex, and New Castle, respectively. If more data were available regarding land use or light levels in Delaware at the county level, that could be leveraged to develop additional disaggregations. Each disaggregation estimation inherently introduces error, but each of them preserves the sum of population in the state.

We can also combine variables by using county level population to disaggregate the state power demand of 11 million(M) megawatt-hours(MWh) into county level power demand. Intuitively, this would be more reasonable than disaggregating uniformly or by area, and yields a power division of 1.95M MWh, 2.44M MWh, and 6.6M MWh between the three counties. Using one variable to disaggregate another by defining necessary ratios in the refinement is a function suitable to many applications. As SIMoN matures, more aggregation-disaggregation pairs will be implemented based on application.

8 SOFTWARE FRAMEWORK

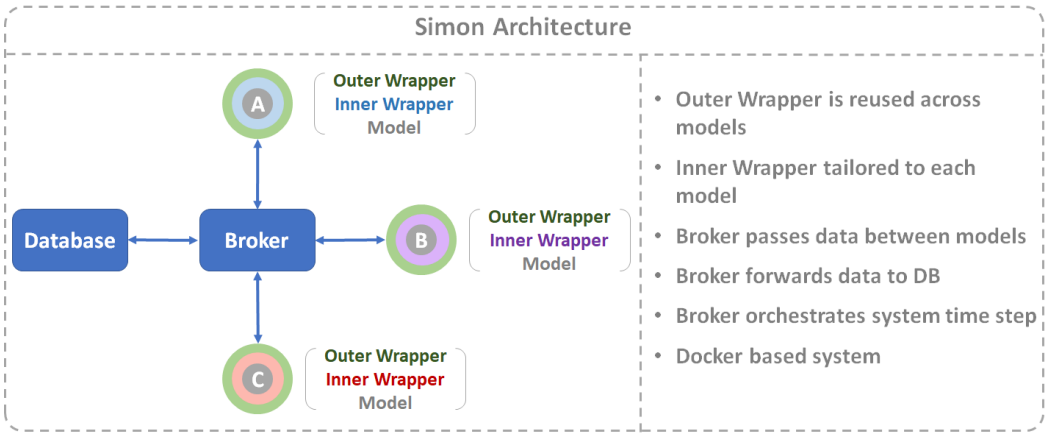

Figure 5: SIMoN Framework Architecture

The SIMoN framework is designed to be modular, extensible, and flexible to a variety of black box models. A diagram of the system architecture is included in Figure 5. Each model runs in a separate Docker container enabling different models in the framework to have conflicting dependencies. A universal outer wrapper operates in each model container to standardize the model interface with the system. This outer wrapper will also perform the data transformations described in Section 7. A central broker, also in its own container, orchestrates model runs and initiates the first time step. After each time step, the models publish their data in JSON format using ZeroMQ. The broker archives these data messages in a backend database for post processing. New models can be introduced into the framework by building a docker image that can execute the model. The user must also construct an inner wrapper which translates the model inputs and outputs into their respective JSON schemas, which include an attribute for geospatial granularity. These granularities are constructed in Section 5. Current SIMoN data transformations are lightweight and run very fast relative to model timesteps, although more complex transformations may be defined in the future.

The Docker-based architecture was selected to enable parallel computing and scaling to HPC, although no extensive scaling experiments have been run to date.

A populated view of four models coupled through SIMoN is contained in Figure 2, where you can also see the inner and outer wrappers drawn around each model output. Open source code for the SIMoN framework is available at https://github.com/JHUAPL/SIMoN.

9 LIMITATIONS AND NEXT STEPS

SIMoN requires that each model run one time step at a time within a Docker container. This assumes that models can be controlled precisely, and that all models/data have the same fixed-increment time progression. In the future, SIMoN framework could be extended to temporal transformations, constructing graphs that capture the interactions of time windows such as days, weeks, months, years, seasons, and billing cycles. This would enable coupling with heterogeneous fixed-increment time steps, but still would not accommodate event-driven simulation.

SIMoN presents a new tool for coupling, but the results of a SIMoN run ultimately depend on the quality of the models/data, the selected coupling architecture, and the amount of error introduced by granularity transformations. For example, early experiments with the SIMoN models detailed here yielded unreasonable results due to an elevated rate of population growth. Future work will extend SIMoN to itself be used as a validation tool, by comparing the results of joint model runs against each other and/or the output of validated models. More advanced backcasting tools could also be added to the system.

The resource models incorporated into SIMoN for test are relatively simplistic, and were primarily used to demonstrate our ability to perform geospatial transformation. We plan to expand the SIMoN Domain Model Library to include more complex fidelic models, additional important components of the FEW nexus such as land use, international models with larger overlapping scopes, and models with more extensive computational requirements to test the limits of SIMoN scalability.

10 CONCLUSION

SIMoN provides new, flexible tools to couple models and data with inconsistent geospatial definitions. We have released a general framework for organizing geospatial interactions, software tools for generating graphs and shapefiles, and an API for joint model runs. Results are shown that leverage these novel coupling mechanisms to connect simple models of population, climate, power, and water systems in the continental United States.

The modular SIMoN framework can be used for a range of experiments comparing models and their interactions. Model perturbations which represent new policies or technologies can be incorporated into the framework easily, ultimately allowing decision makers to better understand the long-term impacts of environmental interventions on resource availability. SIMoN could also be extended to other domains where geospatial interactions interfere with data and model assimilation.

ACKNOWLEDGMENTS

The authors wish to acknowledge support from the Research and Exploratory Development Mission Area of the Johns Hopkins University Applied Physics Laboratory. The authors would like acknowledge the JHU/APL CIRCUIT program and its organizers for funding support, training and coordination of the students on this project. Additionally, we thank Daniel Fiume and Praagna Kashyap for their help with background research and model implementation. We would also like to thank Cytoscape, the graph visualization tool used for the diagrams presented in this paper.

REFERENCES

Bazilian, M., H. Rogner, M. Howells, S. Hermann, D. Arent, D. Gielen, P. Steduto, A. Mueller, P. Komor, R. S. Tol et al. 2011. "Considering the energy, water and food nexus: Towards an integrated modelling approach". *Energy policy* 39(12):7896–7906.

Bussieck, M. R., and A. Meeraus. 2004. "General algebraic modeling system (GAMS)". In *Modeling languages in mathematical optimization*, 137–157. Springer.

Dempsey, M. 2006. "Dymola for multi-engineering modelling and simulation". In *2006 IEEE Vehicle Power and Propulsion Conference*, 1–6. IEEE.

Edmonds, J., and J. Reiley. 1985. *Global energy-Assessing the future*. Oxford University Press, New York, NY.

Edmonds, J., M. Wise, H. Pitcher, R. Richels, T. Wigley, and C. Maccracken. 1997. "An integrated assessment of climate change and the accelerated introduction of advanced energy technologies-an application of MiniCAM 1.0". *Mitigation and adaptation strategies for global change* 1(4):311–339.

Endo, A., I. Tsurita, K. Burnett, and P. M. Orencio. 2017. "A review of the current state of research on the water, energy, and food nexus". *Journal of Hydrology: Regional Studies* 11:20–30.

Giampietro, M., K. Mayumi, and J. Ramos-Martin. 2009. "Multi-scale integrated analysis of societal and ecosystem metabolism (MuSIASEM): Theoretical concepts and basic rationale". *Energy* 34(3):313–322.

Griffies, S. M., M. Winton, L. J. Donner, L. W. Horowitz, S. M. Downes, R. Farneti, A. Gnanadesikan, W. J. Hurlin, H.-C. Lee, Z. Liang et al. 2011. "The GFDL CM3 coupled climate model: characteristics of the ocean and sea ice simulations". *Journal of Climate* 24(13):3520–3544.

Hermann, S., H. H. Rogner, M. Howells, C. Young, G. Fischer, and M. Welsch. 2011. "In The CLEW Model–Developing an integrated tool for modelling the interrelated effects of Climate, Land use, Energy, and Water (CLEW)". In *6th Dubrovnik Conference on Sustainable Development of Energy, Water and Environment Systems*.

Hyndman, R.J. and Athanasopoulos, G 2018. "Forecasting: Principles and Practice, 2nd edition".

Lee, A., J. Sieber, and C. Swartz. 2005. "WEAP. Water Evaluation and Planning System. Userguide. Stockholm Environment Institute". *Tellus Institute, Boston, MA*.

Lubega, W. N., and A. M. Farid. 2014. "Quantitative engineering systems modeling and analysis of the energy–water nexus". *Applied Energy* 135:142–157.

Michigan Water Resources Division 2010, Mar. "General Guidelines for Calculating a Water Budget". retrieved from Michigan Water Resources Division, https://www.michigan.gov/documents/deq/wrd-water-budget_565040_7.pdf.

Smith, C. J., P. M. Forster, M. Allen, N. Leach, R. J. Millar, G. A. Passerello, and L. A. Regayre. 2018. "FAIR v1. 3: a simple emissions-based impulse response and carbon cycle model". *Geoscientific Model Development* 11(6):2273–2297.

Tidwell, V. C., P. H. Kobos, L. A. Malczynski, W. E. Hart, and G. T. Klise. 2009. "Decision Support for Integrated Water-Energy Planning.". Technical report, Sandia National Lab.(SNL-NM), Albuquerque, NM (United States).

United States Census Bureau 2018, Mar. "Population, Population Change, and Estimated Components of Population Change: April 1, 2010 to July 1, 2018". retrieved from United States Census Bureau, https://www.census.gov/data/tables/time-series/demo/popest/2010s-counties-total.html.

Unites States Geographical Survey (USGS) 2019. "National Hydrography Products". data retrieved from USGS, https://www.usgs.gov/core-science-systems/ngp/national-hydrography/access-national-hydrography-products.

U.S. Energy Information Administration 2019a. "Form EIA-861 Annual Electricity Sales". data retrieved from U.S. Energy Information Administration, https://www.eia.gov/electricity/data/eia861/.

U.S. Energy Information Administration 2019b. "Form EIA-923". data retrieved from U.S. Energy Information Administration, https://www.eia.gov/electricity/data/eia923/.

Yates, D., and K. A. Miller. 2013. "Integrated Decision Support for Energy/Water Planning in California and the Southwest.". *International Journal of Climate Change: Impacts & Responses* 4(1).

AUTHOR BIOGRAPHIES

MARISA HUGHES graduated from Binghamton University with a B.S. in Mathematics in 2005. Marisa earned her Ph.D. in Mathematics from Cornell University in 2012. After completing her Ph.D., she worked as a Visiting Assistant Professor at Hamilton College for one year. She then moved to the Johns Hopkins Applied Physics Laboratory in Laurel, MD, in 2013. She works on research in graph theory, data science, systems analysis, and model integration. Her email address is Marisa.Hughes@jhuapl.edu.

MICHAEL KELBAUGH studied Computer Science (B.S.) and Economics (B.A.) at the University of Maryland, Baltimore County, and is currently studying Data Science (M.S.) at Johns Hopkins University. He joined the Johns Hopkins University Applied Physics Laboratory as a software engineer, where his research interests include artificial intelligence, information operations, and the application of technical analysis to public policy. He is a member of the Phi Beta Kappa honor society. His e-mail address is Michael.Kelbaugh@jhuapl.edu.

VICTORIA CAMPBELL studied Mechanical Engineering at Cornell University in Ithaca, NY, and earned a B.S. in 2016 and an M.E. in 2017. Her current role as a mechanical engineer at the Johns Hopkins University Applied Physics Laboratory is focused on mechanical design and finite element analysis, with a wide range of work in defense and space. Her master's

work was focused on renewable energy systems and her current research interests lie in modeling and simulation and advanced prototype development. Ms. Campbell is a member of the Tau Beta Pi engineering honors society and the Society of Women Engineers. Her email address is Victoria.Campbell@jhuapl.edu.

ELIZABETH REILLY was born in Washington, DC. She studied mathematics at Wake Forest University (B.S.) in Winston-Salem, NC, and the University of South Carolina (M.A.) in Columbia, SC. She received her Ph.D. in Applied Mathematics and Statistics with an emphasis in graph theory at the Johns Hopkins University. After graduate school, she joined the Johns Hopkins University Applied Physics Lab where she is currently is a supervisor and senior researcher in the Artificial Intelligence Group focused on next generation Intelligent Systems. Her email address is Elizabeth.Reilly@jhuapl.edu.

SUSAMA AGARWALA studied Mathematics and Physics at MIT, and Economics at the University of Nottingham. She received her Ph.D. in 2009 from Johns Hopkins University, studying mathematical physics, number theory, and combinatorics. After graduate school, she held research positions in mathematics at Oxford University and the University of Hamburg, and faculty positions in mathematics at Caltech, the University of Nottingham, and the US Naval Academy. She joined the Johns Hopkins University Applied Physics Lab as an applied mathematician, where she conducts research as an economic modeler into the foundations of Machine Learning and in understanding the structures of networks naturally occurring in connectomics, or brain graphs. Her email address is Susama.Agarwala@jhuapl.edu.

MILLER WILT completed his bachelor's degree in computer engineer at Lehigh University (Bethlehem, PA) in 2014, and his master's degree in computer science at Johns Hopkins University (Baltimore, MD) in 2018. After finishing his undergraduate degree, he began working at the Johns Hopkins University Applied Physics Lab as a Software Engineer. Starting in 2018, he transitioned to being a Research Engineer at the lab, focusing on machine learning research and development. His email address is Miller.Wilt@jhuapl.edu.

ANDREW BADGER received his bachelors in computer engineering with a minor in mathematics from University of Maryland College Park in 2014 and then received his master's degree in applied biomedical engineering from Johns Hopkins University in 2019. He has worked as a software engineer at JHU/APL since 2015. He has contributed to efforts in robotics, machine learning, and mixed reality within JHU/APL's Intelligent Systems Center. His email address is Andrew.Badger@jhuapl.edu.

EVAN FULLER received a B.S. in Mathematics from the University of Texas at Austin in 2004, followed by a PhD in combinatorics from The University of California, San Diego in 2009. Following that, he worked as a Professor at Montclair State University in New Jersey. Currently, he is a data scientist with the United States Department of Defense in the greater Washington, D.C. area. His email address is fuller.evan@gmail.com.

XIMENA CALDERON AREVALO is from Fullerton, CA. She is a junior studying Applied Mathematics and Statistics at the Johns Hopkins University. She began interning at the Johns Hopkins Applied Physics Laboratory in May of 2019 with the CIRCUIT Program. Her interests include optimization and data analysis. Her e-mail address is ximenacalderon0917@gmail.com.

ALEX FIALLOS is a graduating senior from Johns Hopkins with a B.S in Biomedical Engineering and Applied Mathematics. He joined the Johns Hopkins University Applied Physics Laboratory as a CIRCUIT intern where his research interests include data science, computer vision, and software development. His e-mail address is afiallo1@jhu.edu.

LYDIA FOZO is studying Chemical and Biomolecular Engineering with a minor in Computational Medicine at Johns Hopkins University. She is an intern at the Johns Hopkins Applied Physics Laboratory under the CIRCUIT program. Her interests include machine learning, data analysis, and personalized medicine. Her e-mail address is lfozo1@jhu.edu.

JALEN JONES is studying Public Health at Johns Hopkins University. She is an intern at the Johns Hopkins University Applied Physics Laboratory under the CIRCUIT program. Her interests include health equity, health surveillance, population demographics, and epidemiology. She is a Bloomberg Scholar. Her e-mail address is jjone264@jhu.edu.

DILLON PONZO attended high school at the Marine Academy of Technology and Environmental Science in Manahawkin, NJ and went on to receive a B.S. in Environmental Engineering at Johns Hopkins University in 2016, as well as an M.S.E. in Computer Science in 2019. He currently works as a software development engineer at Amazon Web Services in Seattle, WA. Previously, he worked as a software developer and researcher at the Intelligent Systems Center of the Johns Hopkins University Applied Physics Laboratory. His email address is dponzo18@gmail.com.