adaptation and modularity, along with other complex systems features such as interactions and behaviors [2]. CASL also requires entities to be defined as either agents or environments. We propose an extension to the language and framework, called CASL-SG, which requires the model designer to restructure their initial model slightly, by considering the collective relationship between entities. Each collective, or group, consists of agents that have a semantic relationship. As such, we refer to these collectives as 'semantic groups'. This relationship is also dependent on how particular agents are represented and in many cases, how agents are represented can form the basis for their relationship. For example, in an emergency department, a patient has a much stronger relationship with a doctor than a nurse or a pathology technician. As collective behaviors and aggregation is a key feature of CAS, defining which entities comprise 'semantic groups' is relatively intuitive. This has previously been achieved for models where the environment is represented by two or three dimensional space such as a flock of birds model or a traffic model, and has relied on forming groups based solely on the agent's position in space.

Once a model is constructed in the CASL modeler, code is generated only if all the required constraints have been adhered to. The generated code is then executed in the simulator, which may require initialization parameters which can be provided by a configuration file. The observation tool is comprised of several modules that analyze various features of a CAS such as emergence, self-organization, adaptability, runtimes, interactions, and domain-specific features. The observation tool is designed to be extensible to allow for new metrics to be added, that may either be designed specifically for the current simulation or for a more domain-agnostic purpose. To aid the observation module, CASL allows for certain model features to be flagged so that these are utilized by a particular metric. For example, an agent may make multiple types of interactions, but only one type is considered useful for study. CASL allows the model designer to flag that particular interaction type for use in an interaction metric.

We have designed and implemented a range of metrics for our observation tool to study key CAS features such as criticality, self-organization, emergence, and adaptability. We have performed extensive experiments using these metrics by analyzing two distinct CAS models, namely an emergency department and a Game of Life model.

3. RESULTS

We have implemented several models in CASL using semantic groups such as Game of Life, Flock of Birds, an Emergency Department, and a social network. Our prototype simulator relies on Repast Simphony [1] for simulation scheduling and our distribution is achieved using parallel processing. For very large models, CASL with semantic groups provides a significant runtime decrease when compared to CASL without semantic groups. A Game of Life simulation with 4 million cells took 28 minutes using a single group , which was reduced to 8 minutes when using 100 semantic groups. A Flock of Birds simulation with 5,000 agents was roughly 18.5 times faster than the non-semantic group equivalent. Furthermore, semantic grouping allowed for executions with more than 10,000 agents to finish, which was not possible using the non-semantic group equivalent. Our significantly more complex emergency department simulation had a speed up of roughly 50% when using 30 semantic groups, compared to one semantic group. We have attained similar speed-ups across our other simulations.

Our metrics were able to detect criticality, self-organization, emergence, and adaptability in several cases of our emergency department and Game of Life models. Further work, including refining existing metrics, and creation of new ones is ongoing. In addition, the observation module of CASTLE allows for external analysis tools to utilize the data generated by a simulation.

4. CONCLUSION AND FUTURE WORK

We propose a framework, CASTLE, for modeling, simulating, and analyzing complex adaptive systems. Our language, CASL, is capable of creating CAS models, while the extension, CASL-SG, is capable of designing models that can be simulated with very large numbers of agents. We are currently using CASTLE to perform an in-depth study of smart cities focusing on transportation networks. We aim to use our observation tool, existing metrics, and other analysis tool-kits to study smart cities from a complex adaptive systems perspective. We focus our analysis on examining emergent behaviors, self-organization, and adaptability. Our future work includes designing new metrics to study more complex properties such as self-similarity and feedback loops, improving the performance benefits gained with CASL-SG by enabling the simulation to utilize GPU acceleration, as well as allowing metrics to be defined using CASL. In addition, we aim to showcase to flexibility and power of CASL-SG by creating models of realistic systems.

REFERENCES

[1] The Repast Suite. `https://repast.github.io/`. Last Accessed: 01/11/2016.

[2] L. Birdsey, C. Szabo, and K. Falkner. CASL: A Declarative Domain Specific Language For Modeling Complex Adaptive Systems. In *Proceedings of the Winter Simulation Conference*, 2016.

[3] W. K. V. Chan. Interaction Metric Of Emergent Behaviors In Agent-Based Simulation. In *Proceedings of the Winter Simulation Conference*, pages 357–368, 2011.

[4] J. H. Holland. Studying Complex Adaptive Systems. *Journal of Systems Science and Complexity*, 19(1):1–8, 2006.

[5] S. Mittal. Emergence In Stigmergic And Complex Adaptive Systems: A Formal Discrete Event Systems Perspective. *Cognitive Systems Research*, 21:22–39, 2013.

[6] M. J. North, N. T. Collier, J. Ozik, E. R. Tatara, C. M. Macal, M. Bragen, and P. Sydelko. Complex Adaptive Systems Modeling With Repast Simphony. *Complex Adaptive Systems Modeling*, 1(1):3, 2013.

[7] Ö. Özmen, J. Smith, and L. Yilmaz. An Agent-based Simulation Study Of A Complex Adaptive Collaboration Network. In *Proceedings of the Winter Simulation Conference*, pages 412–423, 2013.

[8] H. V. D. Parunak and S. Brueckner. Entropy and Self-Organization in Multi-Agent Systems. In *Proceedings of the International Conference On Autonomous Agents*, pages 124–130, 2001.