Application of Proven Parallel Programming Algorithmic Design to the Aggregation/De-aggregation Problem

Thomas D. Gottschalk
Center for Advanced Computing Research, Caltech
Pasadena, California
tdg@cacr.caltech.edu

Dan M. Davis
Information Sciences Institute, Univ. of So. Calif.
Marina del Rey, California
ddavis@isi.edu

ABSTRACT

A continuing problem in entity-level, intelligent agent simulations has been one of efficiently, effectively and expediently aggregating smaller units like squads and platoons into larger ones like companies and battalions and then de-aggregating them again at appropriate times. This paper reviews the goals and issues of the aggregation/de-aggregation (A/DA) problem and then lays out some solutions based on High Performance Computing, computational science and lessons learned from advanced techniques, such as adaptive simulation meshes. Experience has shown and logic dictates that aggregation is a more straightforward operation than is de-aggregation. A/DA of collective units is required for future, large-scale simulations, *e.g.* Sentient World Simulation. Understanding how to distribute the smaller units and how to represent the impacts of the simulation on these segments has largely eluded the M&S community for years. This problem is made more complex by the existence of significant amounts of "legacy code" and this paper gives examples of a successful approach to working with such code in an HPC environment. Three workable solutions are enabled by HPC: simulating all levels continuously while displaying only the designated unit level, simulating smaller entities' behavior with reduced behavioral resolution to save compute resources, and foregoing all lower level simulation by simulating only the top-level designated. This last method requires laying down the lower-level entities using doctrine, status, and terrain to achieve realistic disposition. This paper will investigate the processes, impacts, and performance of all three methods. Entity migration across various compute nodes in cluster computers and germane HPC examples from similar computational approaches will be described. The approach applies methods, shown to be effective in on-going research in the physical sciences, to problems facing the DoD M&S community. Performance analyses are anticipated, as are user evaluations by operators, controllers, and analysts.

ABOUT THE AUTHORS

Thomas D. Gottschalk is a Member of the Professional Staff, a Senior Research Scientist at the Center for Advanced Computing Research (CACR), and Lecturer in Physics all at the California Institute of Technology. He has worked at CACR for nearly a decade advancing the use of massive parallel computers for simulation. His instructional duties include Statistics and Experimental Design for Caltech Physics Graduate students. Dr. Gottschalk has been active in parallel programming for nearly two decades, with efforts spanning integrated circuit design, intelligent agent simulations, theater missile defense, and physics modeling. He consults for a number of other organizations, including his work on space based systems for the Aerospace Corporation. He received a B.S. in Physics from Michigan State University and a Ph.D. in Theoretical Physics from the University of Wisconsin.

Dan M. Davis is the Director, JESPP Project, Information Sciences Institute (ISI), University of Southern California, and has been active in large-scale distributed simulations for the DoD. While he was the Assistant Director of the Center for Advanced Computing Research at Caltech, he managed Synthetic Forces Express, a major simulation project. Prior to that, he was a Software Engineer on the All Source Analysis System project at the Jet Propulsion Laboratory and worked on a classified project at Martin Marietta, Denver. An active duty Marine Cryptologist, he recently retired as a Commander, USNR, Cryptologic Specialty. He has served as the Chairman of the Coalition of Academic Supercomputing Centers and the Coalition for Academic Scientific Computation. He received a B.A. and a J.D., both from the University of Colorado in Boulder.

Application of Proven Parallel Programming Algorithmic Design to the Aggregation/De-aggregation Problem

Thomas D. Gottschalk
Center for Advanced Computing Research, Caltech
Pasadena, California
tdg@cacr.caltech.edu

Dan M. Davis
Information Sciences Institute, Univ. of So. Calif.
Marina del Rey, California
ddavis@isi.edu

I. INTRODUCTION

Aggregation/De-aggregation is a topic with an extended and expansive "paper trail" in terms of documents, proposed frameworks, *etc*. This extensive literature base may, however, have as much to do with the ambiguities and possible interpretations of the term "Aggregation/De-aggregation" as with anything. There are many insights and a plethora of differing, novel approaches contained in the publications.

Much of the earlier work focused on the desirability to aggregate smaller units to conserve precious computing capacity. (Nicol, 1987). In this paper, the authors demonstrate there are other pressing problems that will require an effective and scalable solution to the aggregation/de-aggregation issue.

To provide a better context and perspective for this work, it is useful to review briefly four samplings from the literature (this list should be regarded as representative; it is hardly exhaustive):

Assessments of Aggregation Dynamics.

This work (Davis, P., 1995) addresses what can perhaps be described as a "physics question":

Is it reasonable and plausible to define and develop a dynamic model for aggregate entities, given a dynamic model for components (higher resolution entities)?

Put differently, do closed aggregate models with reasonably bounded numbers of additional parameters even exist. Davis essentially answers with "It depends". Working within the context of Lanchester models for combat sectors, it is shown that the ability to construct/define a high level aggregate from components depends on a number of additional factors:

- High-level strategy
- Overall command and control procedures.
- Relative durations of time scales for different resolution components.

Put differently, there are for DoD applications many ways in which an aggregate level entity is significantly different from merely the sum of its parts. The assumption that aggregation/de-aggregation is plausible (requiring merely some technical hurdles) is, in fact, an assumption that should be questioned.

As an historical note, this is not a problem in some scientific applications, such as John Salmon's work on colliding galaxies (Salmon, 1991). Alas, we again conclude that physics is significantly simpler than "people".

Base Object Model Approach

Gustavson and collaborators (Gustavson, 2004) have developed and proposed a framework for incorporating and/or extending aggregation and De-aggregation capabilities within standard (*e.g*., HLA Federations) simulations (Gustavson, 2001). The introduction and the rationalization for this work include a somewhat standard appeal to limited computing resources as one driver for Ag/De-Ag within simulations. That is, the drivers behind the suggested formalism are essentially resource constraints, and it is apparently assumed that a flexible formalism for switching between high and low resolution components is the essential requirement. Quoting from [Gustavson, 2004]:

"BOMs (Base Object Models) have been specifically identified as a potential facilitator for providing reusable object model components to be used for the rapid construction and modifications of simulations and simulation spaces. … Essentially, the coupling of interface (IF) BOMs – known as "mega-BOMs" is intended to define higher order patterns …"

There is a fair amount of reasonable/good computer science work in this approach, particularly within an HLA-based culture. However, it seems to us that this emphasis on the computational structure of Ag/De-

Ag is a bit premature. The issues should be driving the implementation, not the other way around.

Issues in Aggregation Modeling

An excellent paper by a team from the University of Virginia, (Reynolds, 1997) surveys a number of straightforward, technical issues identified in "conventional" aggregation/de-aggregation approaches. Including:

Teleportation: Rapid [High]=>[Low]=>[High] transitions can make the "reconstituted" high resolution entities appear to blink/teleport.

Chaining: Localized entity-level interactions can cause wide-scale De-aggregation – involving far more entities than those involved in the specific entity-entity interaction.

Our perspective – these are really phantom issues, resulting from what might be called "model switching" in which the non-operating model (*e.g.*, HiFi) is discarded and then somehow restarted when the other (LoFi) model is finished. This is not necessary! Instead, we will propose putting HiFi into a "dormant but not dead" state, in which entities are propagated along individual, low-res trajectories with the intent and result of maintaining Hi⇔Low consistency.

Multi-resolution, Multi-perspective Modeling (MRMPM)

There is an excellent overview (Bigelow, 2003) of essential issues in modeling large, complex situations with simultaneous, overlapping component models covering a variety of resolutions/fidelities. It is tempting (to us, at least) to summarize key claims from this work as follows:

- Neither HiRes nor LowRes models, by themselves, are adequate. Put differently:
  - Ag/DeAg is not primarily a concession to limited computing resources.
  - Indeed, the real issues are not solvable by buying more computers.
- HiRes and LowRes models describe different components of an overall, complex situation. In a sense, neither can be "subordinate" to the other. In the simplified platoon ⇔ battalion ⇔ company world, the actions of HiRes entities must reflect overall goals and strategies that only make sense within the LowRes world.
- Model uncertainties (stochastics) are essential – consider the number of HiRes configurations that will map onto a single LowRes state.
- Validation is essential to the entire picture – from the outset!

This paper can perhaps be viewed as a modest, initial attempt towards implementation strategies for "Aggregation/De-aggregation" that recognize and respect the importance of the different resolutions and perspectives afforded by simultaneous, multi-resolution modeling.

Put differently, "aggregation" is not something that is done solely to save CPU cycles. Rather, it is needed as a simultaneous, egalitarian component in order to ensure that the HiRes entities are properly positioned in relation to the environment, disposed in accordance with rational rules, reflect the state appropriate to the state of the aggregated entity and are capable of maintaining entity coherence in position and state.

Overview: Definitions and Current Practice

Simulations view collective entities in different ways. In the simplest case illustrated by the cartoon in Fig.(1), a "Platoon" could be viewed as either a single entity or as a collection of four component entities.
The first issue has to do with the semantics of the two representations. In particular, what are the "state variables" used to describe the aggregate. Several obvious elements come to mind:

- Number of aggregated entities.
- Health/status as a unit.
- Location, direction/motion of unit as a whole.
- Dispersion of the unit.
- Objectives (C2) of unit as a whole.

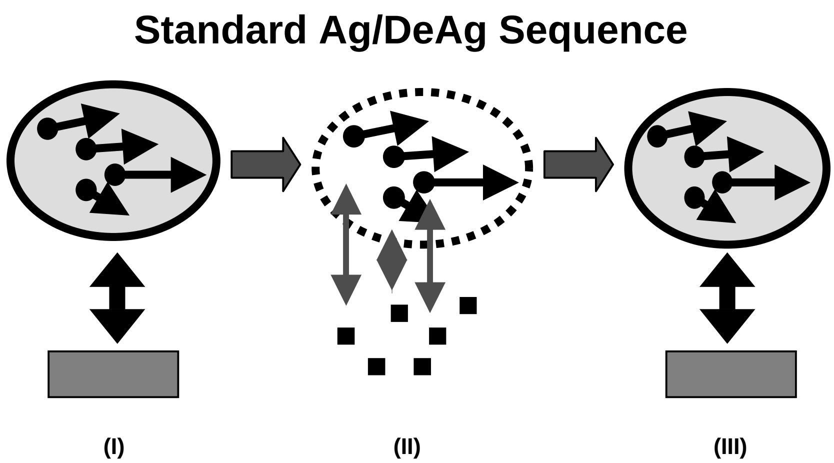

Figure 1: Schematic Ag/De-Ag Configuration

The nominal/notional switching between resolutions are set forth by this picture. That is: I) Initial modeling is done by aggregates, but II) Aggregate encounters situation requiring lower echelon entity-level simulation, requiring time and a sophisticated algorithm to break the aggregate into components, then III) After the triggering event has been satisfied, it is possible to consider a return to the aggregate-level simulation. In Figure 1, the shaded circle might represent a squad, while the arrows might represent individual soldiers. The dark-shaded rectangles might represent an insurgent cell, with the darker-shaded squares representing individual terrorists.

There are some characterizations that apply at the level of the fundamental simulation entities at either level of resolution:

1. Health/status of entity
2. Current position/direction
3. C2 of individual entity.

Two obvious issues arise in this picture:

Translation: Certain characteristics, such as C2, require some level of simultaneous considerations of both resolution levels.

External interactions: Models for Platoon⇔Platoon and Tank⇔Tank interactions are quite different, and switching back and forth between the two can, on the face of it, lead to a variety of problems (“teleportation”, “chaining”, etc.)

II. THE AUTHORS’ PERSPECTIVE

First off, let’s object to a claim made in Gustavson *et al*.: “A key approach to managing large scale simulation exercises and improving overall performance is to aggregate multiple models and entities into inclusive groups”. We don’t agree with this motivation for aggregation. Rather:

Advances in scalable computing (SPPs) and what is nominally known as “Grid Computing” make the managing problems solvable – if not downright moot.

There are fundamental reasons to <u>require</u> both resolutions in order to produce a meaningful, valid, and useful simulation. The example here is the C2 component – C2 decisions are, by their nature, functions of various levels of aggregate states, and consistent representations of <u>both states</u> are required in order to have C2 decisions/directives that reflect the overall simulation state.

Rather than switching back and forth between different resolutions, both resolutions can and should be evolved within a single, overall simulation framework, as described in the next subsection.

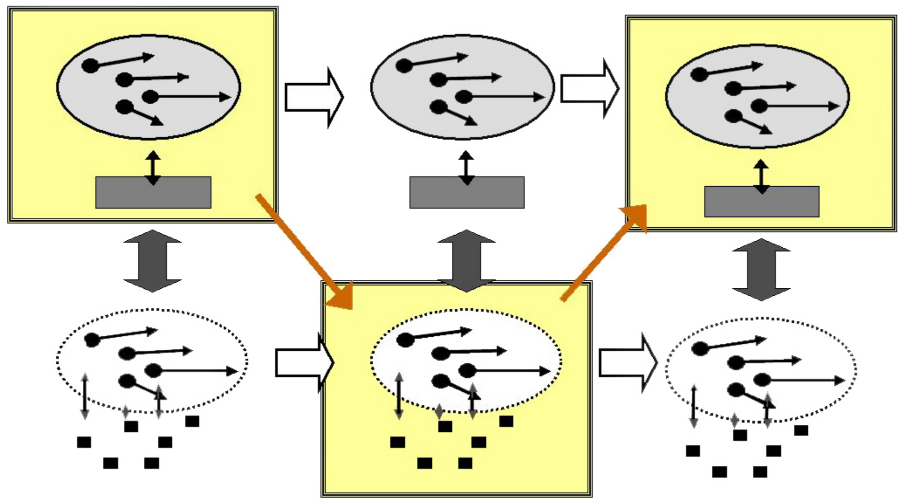

Figure 2: Multiple resolutions with alternating control (the SCAR picture).

We suggest that SPPs and general metacomputing provide a scalable, general solution to the high level problems in conventional Ag/DeAg approaches. A high-level description of the SPP approach is shown in Fig.(2). The high-level components in this strategy are as follows:

- Both representations are maintained at all times within the simulation.
- At any point in time, one and only one mode is in control (the shaded regions in the figure).
- The dominant/controlling simulation mode is responsible for updating relevant object data for the other representation/mode.

The third point requires some clarifications. If the high-resolution (tank) picture is in charge, it is relatively straightforward to update collective parameters of the aggregated state. The other direction is, of course, harder. Our suggestion here is based on a number of straightforward observations:

- A "collective state" description is a representation of an ensemble of distinct, entity-level states and trajectories.
- Much of the computational complexity of the entity-level management is done by selecting a single representative trajectory from the ensemble and propagating it. (This is arguably not much different from the old SAF remote vehicle tables)
- Changes of state in the collective picture can thus be matched with selections of an arbitrary representative trajectory within the consistent ensemble.

Consistency of perceived state, however, is not the only reason for proposing simultaneous simulations at multiple resolutions. The benefit of this "stereo resolution" is the possibility of consistent modeling of "scale transcendent" components, such as C2.

This last point is extremely important, and it is worthwhile to restate it rather bluntly:

- Each representation/perspective (HiRes/DeAg versus LoRes/Ag) contains information that is simply *not present* in the other representation.
- Neither representation, by itself, is adequate for addressing the objectives of FMS.

Consider a particular high-level analysis objective: the identification of failure modes for a plausible force deployment strategy. The aggregate model is needed to explore basic C4ISR concepts. However, the success or failure of any individual element in the high-level strategy is ultimately a probabilistic issue, dependent on the outcomes of individual entity-level interactions.

It is the authors' experience and also is putatively well accepted that probabilistic/aggregate approaches are inadequate for assessing failure probabilities and failure modes, *e.g.* in simulations of space-based IR surveillance networks. Correct probability estimates require higher fidelity, higher resolution simulations of relevant components.

The components/concepts in our proposed SCAR architecture (Fig. (2)) can be stated as follows:

- Both HiRes/DeAg and LoRes/Ag simulations are implemented and run simultaneously.
- At any point in time, one picture is “dominant” (yellow components of Fig. (2)) while the other is “subservient”
- Simple models/procedures maintain a representative subservient state consistent with predictions of the dominant representation.

A key to this approach is the coupling of “equations of motion” implicit in the third point. The manner in which this is done depends on which of the two representations is dominant:

HiRes (Entity) Dominant: The collective state variables for the aggregate are periodically recomputed and updated from the entity-level states. The aggregate model receives but does not alter these parameters (“input only” mode).

LoRes (Aggregate) Dominant: We assume that the aggregate state in fact represents a statistical ensemble of plausibly associated entity-level configurations. When the aggregate representation assumes dominance, a single representative from this ensemble is selected. This ensemble is propagated using simplistic models as long as the aggregate model remains dominant.

The simplified entity propagation envisioned here is similar to that already done for “remote vehicles” within standard SAF models. The entity states are maintained using simple, constant velocity trajectories, and “subtleties” such as road boundaries and possible obstacles (trees, buildings, …) are simply ignored. The details of the subservient entity state may be questionable, but the collective state maintains consistence with that predicted/driven by the dominant aggregate model.

Note that the overall SCAR approach requires consistent, sometimes frequent coupling of the parallel representations from Fig.(2).

When the HiRes/Entity view is dominant, a number of “global combine” operations are needed to update current collective parameters as assumed/needed in the aggregate model. This is “standard stuff” for distributed computing.: Salmons colliding galaxy calculations (Salmon, 1997).

When the LoRes/Aggregate representation is dominant, the maintenance of the HiRes configuration is a bit more convoluted:

- The entity state at the time when the aggregate view assumes dominance selects the appropriate representative from the entity-level ensemble consistent with the aggregate
- Entities are generally propagated using simple models (“dead reckoning”).
- Changes in aggregate-level quantities (e.g., unit health) are implemented by random selections (e.g., pick any two tanks from a platoon and disable them.)

This too, we assert, is not a difficult task or concept within standard, distributed computing practices. Moreover, the required couplings between dominant and subservient representations are simplified when one realizes (as in Bigelow and Davis) that the entity and aggregate simulations operate on very different time scales, so that the very large scale communications/redistributions associated with larger aggregation levels are needed far less often.

III. A TEST BED FOR SCAR DEVELOPMENT

There are clearly many issues in multiple dimensions that must be addressed in converting the general observations of the preceding section into an actual software system that addresses real issues. As experience in High Performance Computing has shown, it is best to begin with small incremental implementations and modifications. It would be prudent to suggest that an ideal test case for the general SCAR schematic of Fig.(2) could be found in the merging of entity level simulation federates of varying sizes with inputs from the highly-aggregated PMESII (political, military, social, economic, information, infrastructure) simulations. The ongoing integration of the SEAS (Synthetic Environment for Analysis and Simulation) into JSAF/JUO simulations is a relevant example. More generally, this type of integration is essential to things like Sentient World Simulation.

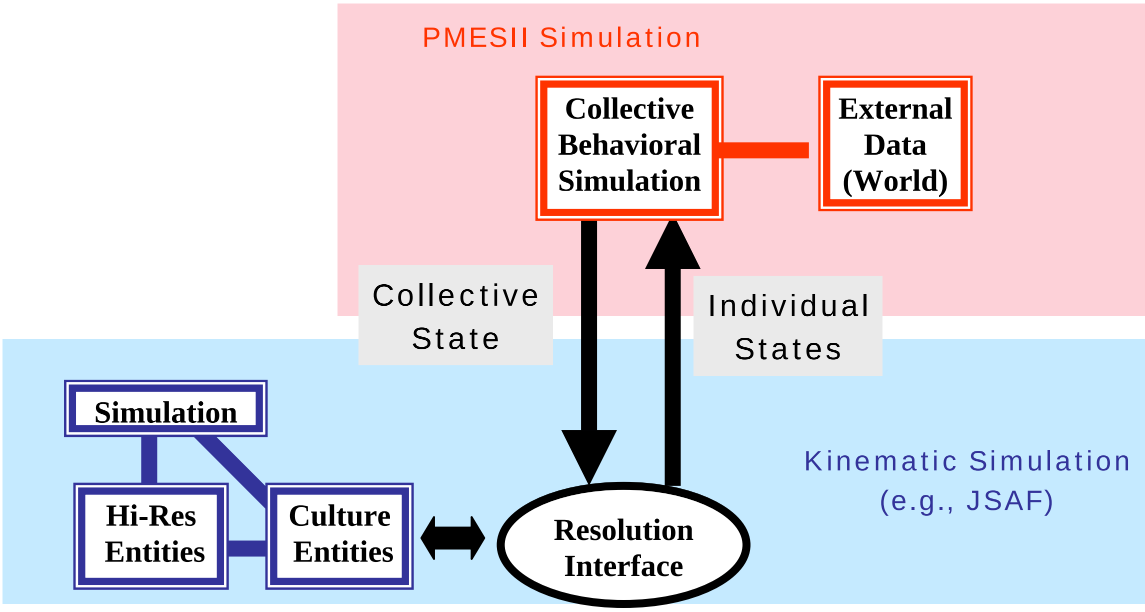

Figure 3: Schematic Behavioral⇔Constructive Integration

Figure 3 illustrates a hybrid simulation of interest, based on ongoing JSAF/JUO activities. The overall simulation can be described in terms of two broad components:

1. An entity-level simulator describing both a number of high-resolution entities and a simulation of a much larger population of lower resolution "background" civilian entities (known as "culture" in standard JSAF parlance).
2. A SEAS higher-level collective/aggregate simulation that models overall behavior of the culture entities, based on a variety of external data.

In order to achieve the goals of the broader JSAF/JUO/CMR investigations, both models are required. Put somewhat bluntly, the individual behaviors for culture entities within JSAF are far too simplified to present a realistic set of inputs for the higher-fidelity entities within the simulation.

In order to provide a more realistic framework/canvas for the simulated high-resolution entities, it is necessary that the collective state descriptions provided by the PMESII component be reflected in the behaviors of individual culture entities. Again, it should be stressed that both models are required. JSAF cannot drive the culture entities towards collective behaviors. PMESII/SEAS does not drive the individual civilian entities needed for a realistic JSAF simulation.

The problem of maintaining consistent civilian/culture representations within SEAS and JSAF is compounded by a practical matter of large-scale distributed simulations – namely the decomposition of the large-scale culture population over a distributed computing resource (e.g., a scalable parallel processor – SPP): In order to optimize the overall simulation (and, in particular, in order to distribute computational resources associated with sensors viewing the civilian population), the culture entities are typically distributed randomly across the individual processors within the SPP. Stating this somewhat more bluntly:

The PMESII simulator deals, significantly, with localized collectives ("crowds").
The entity simulator has little/no connection with any crowd concept.

A consistent, multi-resolution implementation must address this issue, as suggested by the schematic "Resolution Interface" symbol in Fig. (3).

It is the considered opinion of the authors that a productive first model/implementation for this resolution interface can, in fact, be constructed rather easily using standard distributed computing techniques. Stated rather briefly:

Actual entity level states within the high-resolution (culture) simulation provide relevant aggregate information (e.g., crowd densities) to the PMESII module.

Collective state changes (e.g., tendencies towards civil disobedience) as predicted by PMESII are transferred to and distributed among the individual culture entities.

Each of these tasks can be done using standard distributed computing techniques.

The bigger hurdles towards a consistent SCARs approach to JFAF/PMESII have little to do with the communications among and interfaces between the hi-res and low-res components. Rather, the real issues are related to the modeling done within the individual components. The road to a first demonstration of an integrated JSAF/SEAS simulation, for example, would require addressing the following steps:

1. First off, the individual culture entities will need to be expanded to include some sort of "internal state" variable – e.g., a "happiness" value on a scale of 1 to 10. That is, the culture entity simulation must depend on collective states inferred by SEAS is the SEAS component is to provide any true added value.

2. More importantly, the behaviors within the culture entities would need to be expanded so that actions, in some sense, reflect the assumed happiness value. This is actually a non-trivial extension of the existing culture entities. A challenge here lies in constructing simplified entity behaviors (so that very large numbers of entities can be simulated) that nonetheless provide reasonable actions for "happy", "depressed", "suicidal", … individual entities.

3. The PMESII component will make its usual deductions on the state (happiness, etc) of the collective culture state, based on both input from the overall simulation and from relevant external data. The primary output of the PMESII component at this stage would be a predicted "happiness distribution" within the collective culture state. This information is sent down from the PMESII to the entity simulator.

4. The entity simulator then (arbitrarily) reassigns happiness values to the individual culture entities so as to reproduce the collective characteristics determined by PMESII. Again, this follows the paradigm for de-aggregation suggested/noted above:

    > The de-aggregation is essentially the selection of a single, representative entity-level trajectory consistent with the global state determined by the aggregate model (PMESII)

5. The modified behaviors are propagated by the individual culture entities. In time, gross aggregate characteristics of the culture (e.g., the local densities of "extremely unhappy" individuals) are sent back to PMESII, providing input for subsequent aggregate-level evolution.

The linking of the two resolutions implied by the overall SCARs paradigm can be done, initially at least, in a very straightforward fashion. Aggregate densities for various traits translate into (localized) probabilities that any particular entity shares the trait in question. It is reasonable for the "Resolution Component" aspect of Fig. (3) to simply modify traits of individual culture entities based entirely on these probabilities. This is, in essence, an example of selection of a single entity-level representative from the ensemble of configurations associated with the aggregate. This "loosely coupled" paradigm is easily implemented, and sufficient to ensure (reasonable) consistency of the two simulation pictures.

IV. What Does SCARs Offer

The JSAF/SEAS test-bed just described provides a simple, initial implementation of the simultaneous consistent representation picture of Fig. (2), with alternating "primary/secondary" roles. The intertwining of the dominant roles in this case is quite straightforward, with a possible implementation as follows:

1. For most of the time, the JSAF and SEAS components operate independently.
2. The JSAF/culture component periodically computes aggregate quantities (e.g., population densities for various population traits) and sends these updated states to SEAS. The aggregations and communications can be done using standard distributed computing techniques.

3. The SEAS component would assume dominance on a more "interrupt driven" model, sending modified population densities to the entity simulator whenever the external inputs to SEAS and the associated modeling algorithms suggest that a significant change of state has occurred.

On receipt of modified aggregate population parameters, individual entity characteristics are modified to reflect the new aggregate state

First off, the individual culture entities will need to be expanded to include some sort of "internal state" variable – *e.g*., a "happiness" value.

More importantly, the behaviors within the culture entities would need to be expanded so that actions, in some sense, reflect the assumed happiness value.

The PMESII component will make its usual deductions on the state (happiness, etc) of the collective culture state, based on both input from the overall simulation and from relevant external data. The primary output of the PMESII component at this stage would be a predicted "happiness distribution" within the collective culture state. This information is sent down from the PMESII to the entity simulator.

The entity simulator then (arbitrarily) reassigns happiness values to the individual culture entities so as to reproduce the collective characteristics determined by PMESII. Again, this follows the paradigm for De-aggregation suggested/noted above:

The De-aggregation is essentially the selection of a single, representative entity-level trajectory consistent with the global state determined by the aggregate model (PMESII)

The modified behaviors are propagated by the individual culture entities, *e.g*., the local densities of "extremely unhappy" individuals are communicated to the system. In time, gross aggregate characteristics of the culture are sent back to PMESII, providing input for subsequent aggregate-level evolution.

Performance of the JUO net has been more than sufficient to support the suggested configurations for SCAR. The inherent scalability of the system should easily support both a JSAF-like simulation federate and a SEAS-like PMESII federate. (Gottschalk, 2005)

At meetings of the user community, in this case, the J9 Experimentation Directorate of the U.S. Joint Forces Command, the operators and analysts uniformly told the authors that the inclusion of PMESII and other multi-resolution federates was not only desirable, but necessary to complete mission tasking from JFCOM. Their stated requirements were the basis of the authors' interest in this problem and should be reflected in similarly disposed simulation and experimentation groups throughout the DoD.

V. CONCLUSIONS

The authors have no doubt that the previous sense of importance of aggregation and de-aggregation will only become more vital in achieving the goals of the DoD simulators in the future. Asymmetric warfare and urban battlespaces will dictate even more use of simulations to prepare the warfighter for combat.

The authors have experienced the general migration of computing utilization from single processor machines to parallel networks of computers and then on to the scalable parallel High Performance Computers of today (Lucas, 2003). From this experience, they can only come to the conclusion that aggregation and de-aggregation will not only become paramount, it must be developed in a way that will not only allow parallel distributed processing, but must additionally exploit and enhance that capability.

VI. ACKNOWLEDGEMENTS

The authors wish to thank the entire J9 staff of the Joint Forces Command and they want to especially thank Rae Dehncke for his unwavering support and enthusiasm about an expansive vision for HPC. This material is based on research sponsored by the Air Force Research Laboratory under agreement number FA8750-05-2-0204. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Research Laboratory or the U.S. Government.

VIII. REFERENCES

Bigelow, J. H,. & Davis, P. K., (2003) Implications for model validation of multi-resolution, multi-perspective modeling (MRMPM) and exploratory analysis. MR-1750, RAND, 2138, Santa Monica, CA

Davis, P.K. (1995). "Distributed Interactive Simulation in the Evolution of DoD Warfare Modeling and Simulation," Proceedings of the IEEE, Vol. 83, No. 8, pp. 1138-1155.

Gottschalk, T. & Amburn, P., 2005, Extending The MeshRouter Framework for Distributed Simulations Proceedings of the Interservice/Industry Training, Simulation and Education Conference, Orlando, FL.

Gustavson, P., & Chase, T., (2004), Using XML and BOMS to Rapidly Compose Simulations and Simulation Environments, Proceedings of the Winter Simulation Conference, Orlando, Florida,

Gustavson, P., *et. al.* (2001) BOM Study Group Final Report. SISO-REF-005-2001. 15 May 2001. Available at http://www.sisostds.org.

Lucas, R., & Davis, D., Joint Experimentation on Scalable Parallel Processors, (2003), in the Proceedings of the Interservice/Industry Simulation, Training and Education Conference, Orlando, Florida

Nicol, D. M., & . Saltz, J., (1987) Principles for Problem Aggregation and Assignment in Medium Scale Multiprocessors, ICASE Report No. 87-39, September 1987

Reynolds, P., Natrajan, A. & Shrinivasan, S., (1997), Consistency Maintenance in Multi-resolution Simulations, ACM Transactions on Modeling and Computer Simulation, Vol. 7, No. 3, July 1997, Pages 368–392

Salmon. J., (1991), Parallel *O(N Log N)* N-body algorithms and applications to astrophysics. In *COMPCON Spring '91, Digest of Papers*, pages 73-78

Salmon, J. K. & Warren M. S. (1997), Parallel, Out-of-Core Methods for N-body Simulations , CACR Technical Report, CACR-135, Center for Advanced Computing Research, California Institute of Technology, Pasadena, California January 1997