MRE: A Flexible Approach to Multi-Resolution Modeling

Anand Natrajan
Dept. of Computer Science, Thornton Hall, University of Virginia, Charlottesville, VA 22901, USA
Email: anand@virginia.edu, URL: http://www.cs.virginia.edu/~an4m/

Paul F. Reynolds, Jr.
Dept. of Computer Science, Thornton Hall, University of Virginia, Charlottesville, VA 22901, USA
Email: reynolds@virginia.edu, URL: http://www.cs.virginia.edu/~pfr/

Sudhir Srinivasan
Mystech Associates Inc., 5205 Leesburg Pike, Suite 1200, Falls Church, VA 22041, USA
Email: sudhirs@mystech.com, URL: http://www.mystech.com/~sudhirs/

Abstract

Multi-resolution representation of simulated entities is considered essential for a growing portion of distributed simulations. Heretofore, modelers have represented entites at just one level of resolution, or have represented concurrent representations in an inconsistent manner. We address the question of the cost of maintaining multiple, concurrent representations. We present a brief overview of our concept of a Multiple Resolution Entity (MRE) and Attribute Dependency Graph (ADG) both originally described elsewhere, and then compare simulation and consistency costs of some approaches, including our own MRE/ADG-based approach, to multi-resolution modeling. The cost analysis presented here is the first known analysis of its type, and will provide a basis for simulation designers to determine the best, and most cost-effective approach to supporting simulation of entities at different levels of resolution concurrently.

1 Introduction

Multi-Resolution Modeling or MRM is concerned with resolving conceptual and representational differences arising from multiple levels of resolution in distributed simulations joined for a common objective. The crux of the problem is accurately simulating an object *and* its constituents concurrently. If an abstract object and its constituents are modeled concurrently, all interactions in overlapping periods of time with the abstraction and its constituents must be accurately reflected at both levels. However, a number of issues, such as temporal inconsistency and chain disaggregation, arise with concurrent models [Nat95, Reyn97]. These problems have occurred in some form in the majority of simulation couplings; no attempt comprises a satisfactory solution to the MRM problem.

We study the costs associated with operating a simulation at multiple levels of resolution. We discuss existing simulation approaches and introduce our own approach based on the notion of a *Multiple Resolution Entity* (MRE) as the basis for addressing the MRM problem in a consistent manner. If there is a way to maintain consistency within an abstraction interacting concurrently at multiple levels, MREs are our recommended approach for capturing it accurately. MREs are offered in response to the largely *ad hoc* approaches that have been pursued in previous attempts to address the MRM problem. MREs enable isolation of consistency maintenance issues, and they enable possible efficiencies (e.g., using *core attributes*, see [Reyn97]).

2 Multi-resolution modeling approaches

The MRM problem has come to be known as the *aggregation/disaggregation problem* in distributed simulations. It has been found that *Low Resolution Entities* (LREs) and *High Resolution Entities* (HREs) create critical consistency issues when the LREs and HREs interact. A common remedy is to dynamically change the resolution of an LRE (or HRE) to match the resolution of other interacting entities. This dynamic change is called *aggregation* (HREs → LRE) or *disaggregation* (LRE → HREs). However, a disaggregated LRE that re-aggregates itself to interact with another LRE and then later disaggregates, may put itself in a state that it could not have otherwise achieved over the same period of time.

Also, this dynamic aggregation/disaggregation approach incurs problems such as chain disaggregation, network flooding, transition latency and mapping problems between levels. Existing solutions meant to solve some or all of these problems leave the MRM consistency problem unresolved.

We describe briefly some general multi-resolution modeling techniques that have been used. A more complete discussion of these and other approaches may be found in [Reyn97, Nat95]. We broadly categorize these various approaches into two classes: static and flexible approaches. The static approaches — Full Aggregation and Full Disaggregation — represent extreme solutions to the MRM problem. Our own framework, based on *Multiple Resolution Entities* (MREs), is a flexible approach. Details of this approach may be found in [Nat95, Nat96, Reyn97]. We draw cost comparisons between the static and flexible approaches, and argue for the adoption of the latter.

2.1 Static approaches

The approaches described in this section are termed *static* because the resolution level at which entities are simulated is fixed when the simulation is constructed. Representation extremes are called Full Aggregation (the lowest-resolution representation) or Full Disaggregation (the highest-resolution representation).

Full Aggregation stipulates that the entities involved shall be simulated only at the lowest level of resolution, i.e., as abstractly as possible. This approach is used in many aggregate-level simulations for systems analysis, policy analysis and decision support under uncertainty [Davis93]. Aggregate-level simulations are typically low-cost and lend themselves to rapid analysis. Also, they help comprehension of broad scenarios and assist in first-order analysis. *Full Disaggregation* involves the complete disaggregation of a low-resolution entity (LRE) into its constituent high-resolution entities (HREs). Full disaggregation is often used when simulation designers require a high level of detail in the simulation for analysis or because humans are “in the loop” and need the high resolution for perceptual reasons.

2.2 Flexible approaches

Flexible approaches permit an entity to switch levels of resolution during the course of the simulation. Often the simulation level of the entity may change due to interactions with other entities at different levels of resolution. The most common flexible approach is *Partial Disaggregation*, which attempts to overcome the main limitations of full disaggregation by disaggregating only a part of an LRE rather than the entire LRE. The internal boundary across which the disaggregation takes place is known as the *firewall*. With static partial disaggregation, entities do not cross the firewall until an aggregation or disaggregation request is issued and can be satisfied. In dynamic partial disaggregation, entities may transition across the firewall repeatedly, participating in both levels, but at different time-steps. Partial disaggregation is used in the `BBS/SIMNET` linkage [Hardy94, Burd95] in which a `BBS` entity that engages a `SIMNET` entity is partitioned such that one part disaggregates and fights a disaggregate-level battle in the `SIMNET` world, while the other part remains aggregated and fights aggregate-level battles in the `BBS` world. Another linkage that employs partial disaggregation is `AIM` [Seidel95]. Partial disaggregation clearly has the potential to control chain disaggregation. However, this potential depends on the feasibility of constructing a partition inside an LRE. The criteria for constructing the partition must be decided carefully to prevent partial disaggregation from degenerating into full disaggregation.

Consider a situation where an HRE requires the attributes of the constituents of some LRE but does *not* interact with the LRE. In this case, disaggregating the LRE to provide the attributes is wasteful, and *Pseudo-Disaggregation* may be used instead. In pseudo-disaggregation, the HRE receives low resolution information from the LRE and *locally* disaggregates the information to obtain the high-resolution information it needs. Clearly, this technique is not applicable when the interaction between the HRE and the LRE is bidirectional (i.e., requiring some sort of modeling of the constituents of the LRE). Further, in case the LRE is required to disaggregate due to other factors, the algorithm used by the HRE to locally disaggregate the LRE must be the same as that used by the LRE to disaggregate itself. Consequently, this solution is not scalable because each HRE may be required to know how to disaggregate every LRE in the simulation. This scheme is employed by the `JPSD` [Calder95], `TACSIM/CBS` [Smith95] and `Eagle/BDS-D` [Stober95] programs.

3 The MRE-based approach

All of the approaches mentioned in Section 2 maintain for a given entity, at any given time, its attributes at only one level of resolution — the level at which the entity is being simulated. This is unsatisfactory because the attributes at other levels are unused or lost. Typically, the entity is “ghosted” at the levels at which it is not simulated. However, ghosting implies a passive reflection of attributes, not a participation at multiple levels. For an entity to participate actively at multiple levels, it must not only be influenced by events from other entities, but also influence other entities with events at multiple levels. An entity that does so could be said to exist at multiple levels of resolution. Our approach to MRM is a general one based on some fundamental observations [Reyn97]. The focus of our approach is maintenance of consistency among multiple levels of resolution. We propose *Multiple Resolution Entities* (MREs) as entities that can maintain internal consistency across multiple, concurrent levels of resolution. MREs reflect a design strategy, *not* an implementation. MREs may either be designed during the construction of a multi-level simulation or may be created by linking together existing simulations with suitable changes made to incorporate the concepts we discuss here. Clearly, our approach is a flexible one because it enables entities to exist at multiple levels concurrently.

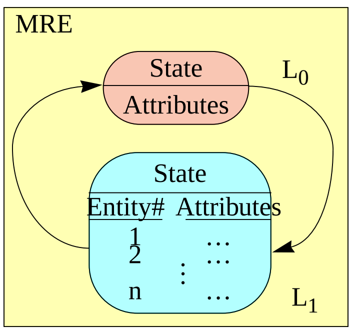


FIGURE 1: Design of an MRE

Each MRE either maintains state information at all desired levels of resolution or furnishes information at a requested level in a timely manner. Simulation of the MRE entails consistently reflecting the effects of interactions at all levels. An MRE interacts at multiple levels of resolution concurrently by internally enforcing logical consistency among corresponding attributes at different levels of resolution. Figure 1 depicts a typical MRE perceivable at two levels of resolution. For the sake of discussion, L_0 is the low-resolution level and L_1 is the high-resolution level. The MRE maintains the attributes at both levels at all times. The two states of the MRE — L_0 and L_1 — are always kept consistent with each other. A more complete description of the MRE can be found in [Nat95, Nat96, Reyn97].

In order to model entity behavior at multiple levels of resolution, relationships among attributes must be captured. These relationships can be modeled by a directed, weighted graph wherein the nodes represent attributes and the edges between the nodes represent relationships. We define the notion of an *Attribute Dependency Graph* (ADG), which depicts the various attributes and sub-entities of the MRE, and portrays the relationships among them. ADGs are an encoding of the concurrent multi-resolution interactions problem, and are also an encoding of solutions thereof. Attributes at all levels are present in the ADG. Therefore, the MRE is represented at all levels of resolution. The dependencies between attributes fall into four classes. The semantics of these dependencies (and hence the edges in the graph) are as below:

- **Interaction dependencies** capture interactions that may cause attributes to change values. Each attribute that can be changed as a direct result of an interaction would have an interaction dependency.
- **Distributive dependencies** are edges from a node representing an aggregate-level attribute to a node representing the corresponding disaggregate attribute for a particular sub-entity.
- **Accumulative dependencies** are edges from a node representing a disaggregate-level attribute for a particular sub-entity to a node representing the corresponding aggregate attribute.
- **Modeling dependencies** are all edges that are not one of the above. Typically, these edges represent relationships between attributes that exist due to the nature of the entity being modeled.

Given an interaction, it is possible to trace a path in the graph to account for changes to nodes.

There exists a spectrum of options to multi-resolution modeling. Options more to the left of the spectrum tend to subsume options to the right. For example, Static Partial Disaggregation, wherein entities lie on one side of the firewall or the other for the entire course of the battle, clearly subsumes Full Disaggregation if we assume that none of the sub-entities in a partially-disaggregated entity lie on the aggregate side of the firewall.

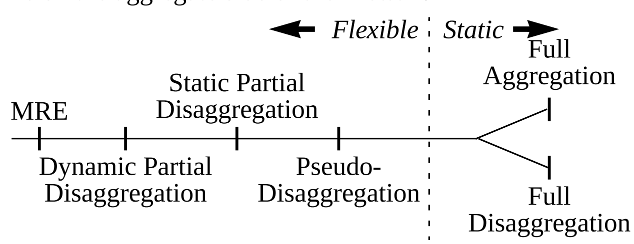


FIGURE 2: Spectrum of options to multi-resolution

The position of the MRE-based approach at the left end of the spectrum is justified because ADGs can be shown to subsume previous approaches. Depending on which level the sub-entity is interacting at, some edges in the graph may be weighted to zero. Some interactions can cause edges between disaggregate-level attributes and aggregate-level attributes to be weighted to zero. If the weights on the edges between all the sub-entity's attributes and the corresponding aggregate-level attributes are zeroed whenever the sub-entity switches to the disaggregate level, then the ADG reduces to dynamic partial disaggregation. This in itself implies that other schemes are subsumed. In addition, if a further decision is made that disallows sub-entities at one level to contribute to another level for the entire course of the battle, we have static partial disaggregation. If we disallow the nodes for a particular level to be created, we have full aggregation or full disaggregation. Thus the ADG is at least as powerful as any traditional scheme. The most significant gains of the flexible approaches lie in the ability to control simulation and consistency costs. As seen in the next section, the MRE-based approach compares favorably to static approaches with respect to balancing simulation and consistency costs.

4 Cost analysis

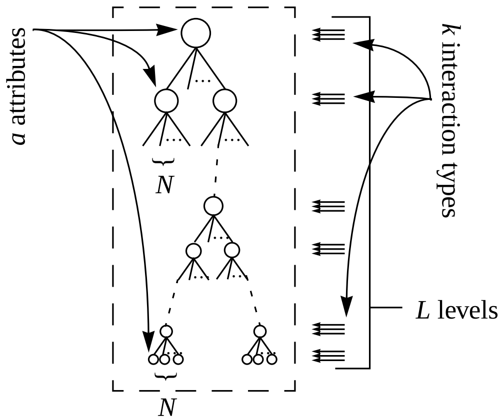


FIGURE 3: Simple Multiple Resolution Entity

It is important to compare the cost of maintaining consistency with the cost of simulation for various techniques of managing multi-resolution simulations. We consider Full Aggregation (FA), Full Disaggregation (FD) and the MRE approach (MRE). FA and FD represent two extreme static solutions, whereas the MRE approach represents a flexible approach. For analysis purposes we use a simplified notion of a multi-resolution simulation. The simplifications merely make it easier to effect a comparison between the various techniques. Figure 3 shows an entity in such a simulation. The assumptions of this simplified multi-resolution simulation are:

- There are L levels of resolution, level 0 being the lowest (most aggregate) and level $L-1$ being the highest (most disaggregate).
- There are N higher-resolution sub-entities per lower-resolution entity, i.e., an entity at a resolution level of i comprises of exactly N entities that are at resolution level $i+1$. This is true for all $i = 0$ to $L-2$.
- All entities at a particular resolution level are exactly identical in composition, i.e., they have the same number of sub-entities (as stated earlier), and also have similar attributes. Note that these entities may perform different tasks in the simulation, but for analysis purposes, they are similar in composition.
- All entities at all levels have exactly a attributes. All the attributes of an entity at a particular level are modified by every interaction at that level.
- There are exactly k types of interactions at each level of resolution.

Therefore,

Total number of entities possible, given a low-resolution entity, $= \Psi(N, L) = \sum_{i=0}^{L-1} N^i = \frac{N^L - 1}{N - 1}$.

Total number of interaction types $= kL$

4.1 Consistency cost

Consistency Cost is comprised of a Static Consistency Cost (SCC) and a Dynamic Consistency Cost (DCC). SCC is incurred during the design phase and is a one-time cost reflecting the amount of effort required to design a consistent entity. DCC is incurred for every interaction at run-time, and reflects the number of operations required to maintain consistency in the face of interactions.

Full aggregation. In FA, an entity is simulated at the

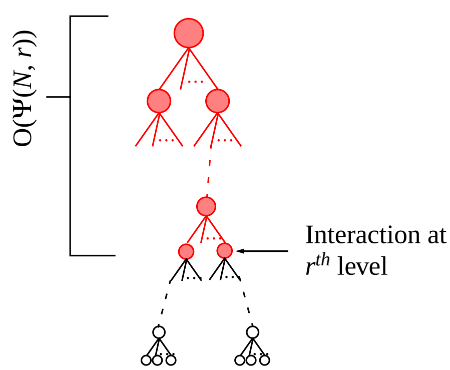


FIGURE 4: DCC for FA

0^{th} level of resolution until a higher-resolution interaction occurs. At that point, the entity is disaggregated to the appropriate level and the effects of the interaction are reflected. In order to maintain consistency, the designer of the simulation has to roll up the effects of the interaction to attributes at all lower levels of resolution. Therefore, each interaction affects $O(La)$ attributes. Assuming all interactions have independent effects (i.e., the combination of the effects of any set of interactions is the same as the effect of the combination of the same interactions),

$SCC_{FA} = O(kL \times La) = O(kL^2a)$

This is the cost of designing functions for reflecting effects of each interaction type on each attribute. However, if we assume that pairs of concurrent interactions could be dependent,

$SCC_{FA} = O(k^2L^2 \times La) = O(k^2L^3a)$

In general, if sets of n concurrent interactions could be dependent,

$SCC_{FA} = O(k^n L^n \times La) = O(k^n L^{n+1} a)$

Assume an interaction at the r^{th} level ($0 \le r < L$) arrives at an entity. The entity must disaggregate to level r, reflect the effects of this interaction at this level and aggregate back to level 0. In order to disaggregate to level r from the current level 0, the costs incurred are $O(\Psi(N, r))$. The cost of aggregation is presumably of the same order as the cost of disaggregation. Thus,

DCC_{FA} (shown shaded in Figure 4) = $O(\Psi(N, r))$

If we assume that the entity does not step through every level between 0 and r during disaggregation/aggregation, then DCC is vastly reduced. However, SCC is then vastly increased because mapping functions must be found for each level so that the entity can "jump" the hierarchy. This optimization not only violates the strict hierarchical nature of the simulation entity, but also may lead to increased inconsistency. This is because the various levels are reachable from one another only through level 0. Thus, they may be inconsistent with each other. In general, FA could cause inconsistency because of the tendency to revert back to level 0, wherein there is a loss of information with respect to higher levels.

Full disaggregation. In FD, all entities are always

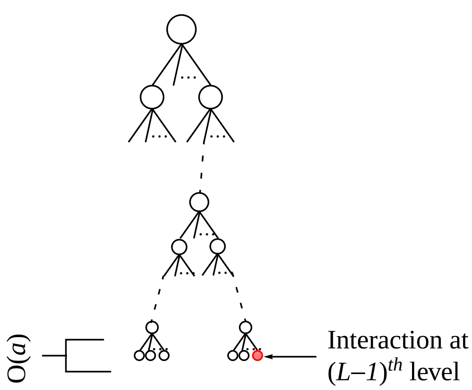


FIGURE 5: DCC for FD

simulated at the $(L-1)^{th}$ level of resolution. Thus, there exists only one level of resolution, namely the highest. Clearly, consistency has to be maintained only within one level, a task far easier than maintaining consistency across many levels of resolution. Therefore, making $L = 1$, each interaction affects $O(a)$ attributes. Assuming all interactions are mutually independent,

$SCC_{FD} = O(k \times a) = O(ka)$

However, if we assume that pairs of concurrent interactions could be dependent,

$SCC_{FD} = O(k^2 \times a) = O(k^2 a)$

In general, if sets of n concurrent interactions could be dependent,

$SCC_{FD} = O(k^n \times La) = O(k^n a)$

The run-time consistency costs for FD are also low. All interactions occur at the $(L-1)^{th}$ level, where $L = 1$. Therefore,

DCC_{FD} (shown shaded in Figure 5) = $O(a)$

The MRE approach. In MRE, an entity is simulated

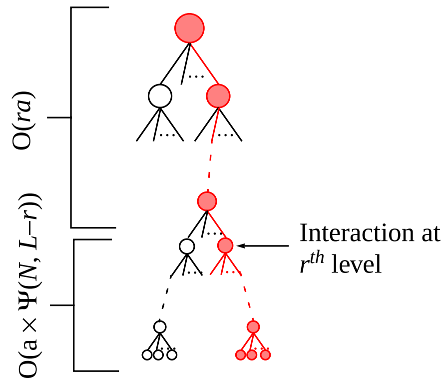


FIGURE 6: DCC for MRE

consistently at all levels of resolution. The relationships between low-resolution attributes and their corresponding high-resolution attributes are well-defined. These relationships — accumulative and distributive dependencies — are essentially the same across levels. In other words, the dependencies between level f and level $f+1$ attributes are much the same as the dependencies between level $f-1$ and level f attributes. The relationships between attributes at various levels are determined without knowledge of the expected interactions on these attributes. Therefore, during the design, each interaction affects $O(a)$ attribute types. The effects are reflected to other attributes by virtue of the pre-set dependencies. Assuming all interactions are mutually independent,

$SCC_{MRE} = O(kL \times a) = O(kLa)$

If pairs of concurrent interactions could be dependent,

$SCC_{MRE} = O(k^2 L^2 \times a) = O(k^2 L^2 a)$

In general, if sets of n concurrent interactions could be dependent,

$SCC_{MRE} = O(k^n L^n \times a) = O(k^n L^n a)$

The run-time costs for MRE are computed as follows. Assume an interaction at the r^{th} level ($0 \le r < L$). The entity must reflect the effects of this interaction at level $= r$, all levels $> r$ and all levels $< r$. In order to reflect the interactions to higher resolution levels, the cost incurred is $O(a \times \Psi(N, L-r))$. The cost incurred in reflecting the effects to lower resolution levels is merely $O(ra)$. Thus,

DCC_{MRE} (shown shaded in Figure 6) = $O(ra + a \times \Psi(N, L-r))$

4.2 Simulation cost

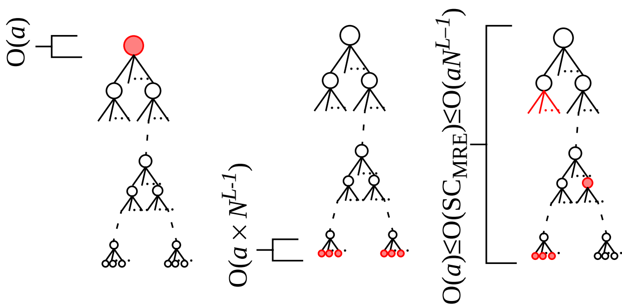


FIGURE 7: (Left to Right) SC for FA, FD and MRE

Simulation Cost (SC) is the cost of simulating the entities during a run of the simulation. SC may include costs of processing, memory and communication. For the purposes of this discussion we will not distinguish between these. For FA, the entity is simulated in the aggregate. Therefore, before disaggregation,

$SC_{FA} = O(a)$

In the case of FD, the entity is always simulated at the $(L-1)^{th}$ level. Therefore,

$SC_{FD} = O(a \times N^{L-1})$

Lastly, SC for MRE lies between SC for FA and FD, because in the worst case, the entity may have to be simulated entirely at the disaggregate level, but in the best case, simulation at the aggregate level may be enough. If there are concurrent interactions at all levels of resolution for an entity, and all the sub-entities at all levels need to be instantiated,

$SC_{MRE} = O(a \times \Psi(N, L))$

However, if there are interactions only at level *0*,

$SC_{MRE} = O(a)$

Figure 7 shows SC for FA, FD and MRE (from left to right) shaded.

4.3 Discussion

TABLE 1: Cost Comparison for various schemes

	SCC	DCC	SC
FA	$O(k^n L^{n+1} a)$	$O(\Psi(N, r))$	$O(a)$
MRE	$O(k^n L^n a)$	$O(ra + a\Psi(N, L-r))$	between
FD	$O(k^n a)$	$O(a)$	$O(aN^{L-1})$

Table 1 compares the various costs for the different schemes considered. Based on this table, Figure 8 shows a rough diagram of expected simulation and consistency costs for FA, FD and MRE.

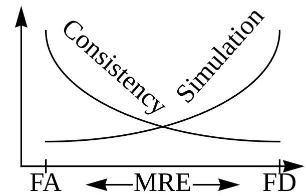


FIGURE 8: Cost diagram

Consistency costs decrease with schemes that run more in the disaggregate. However, simulation costs increase. A scheme running mostly in the aggregate has low simulation costs, but high consistency costs because aggregation tends to cause information loss. The MRE scheme lies between these extremes of multi-resolution schemes. In other words, $SC_{FA} \leq SC_{MRE} \leq SC_{FD}$; $DCC_{FA} \geq DCC_{MRE} \geq DCC_{FD}$.

It is worthwhile to note that in a pathological case DCC_{MRE} may be slightly more than DCC_{FA}, though they both will be of the same order. This is because the consistency gained by the MRE scheme is actually better than the consistency gained by the best FA scheme. FA causes information to be lost when it reverts back to level *0*. This is avoided in the MRE scheme. Likewise, in another worst case, SC_{MRE} may be slightly more than SC_{FD}, though of the same order. However, this is also justifiable in the light of MRE being able to process interactions at all levels, whereas FD is unable to accept interactions at any level except the most disaggregate.

Note the nature of $\Psi(N, L)$. Ψ is polynomial in N, but exponential in L. Since the exponential function grows faster than the polynomial one, it is recommended that for simulations with a flexible object hierarchy, effort should be directed towards making the resolution tree as broad and shallow as possible.

The effects of relaxing the introductory assumptions follow. Clearly, changing L has the most dramatic effect in changes to simulation and consistency costs. Changing N has less dramatic effects. If N is different for entities at different levels of resolution, the function Ψ becomes somewhat involved, but its basic nature does not change. If k changes with the level of resolution, then the total number of types of interactions becomes $k_0+k_1+\ldots+k_{L-1}$ instead of kL. Likewise, we could further complicate the calculations involving the number of attributes, a, by asserting that the attribute count at different levels of resolution is different. None of these modifications to the initial set of assumptions change the order of the costs; they merely make the equations more intricate.

4.4 Practical look at cost analysis

We conducted an experiment wherein we created a mock simulation to measure and compare SC (simulation cost) and DCC (consistency cost[1]) for our three example schemes — FA, FD and MRE. FA is the traditional aggregation-disaggregation approach in which the entity is simulated in an aggregated mode: when a disaggregate-level interaction occurs, the entity disaggregates to the appropriate level, reflects the effects of the interaction and re-aggregates. FD is the approach in which every entity is simulated at the highest resolution level. If interactions at a lower level of resolution occur, they must be somehow translated to equivalent interactions at the highest level of resolution. We chose a particular instance for our MRE scheme where the entity is completely instantiated at all resolutions at all times. Interaction effects are then propagated up and down the hierarchy of entities. We found the MRE-based scheme to hold a lot of promise in terms of being the cheapest approach to CM.

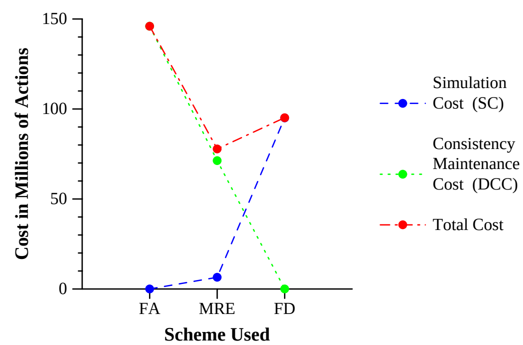
We computed DCC and SC relative to a single entity. Therefore, our pseudo-simulation consists of exactly one Level 0 entity. The interactions are not concurrent because we measured merely the cost of simulation and consistency maintenance, not the quality of consistency. Interactions were described solely by the level at which they occur. The interactions were uniformly distributed across the levels at which the entity was represented. At regular intervals, a special "checkpoint" interaction was injected into the stream of interactions. This interaction serves to tell the entity that it must make itself known to the rest of the simulation, which is akin to sending out periodic entity state messages, as is commonly done in distributed interactive simulations [Dah95].

We calculated SC as the cost of simulating the various entities in the simulation. This entails maintaining the behavior of the entity and performing other tasks to keep the entity a participant in the simulation. We calculated DCC as the cost of maintaining consistency among the many levels of resolution given that interactions at multiple levels occur. All costs were measured by the number of actions required to perform the corresponding task. Initially, all actions were weighted equally. Accordingly, SC_{FA} is the number of actions to perform the checkpoint interaction (always 1) plus the number of actions required to reflect the effect of an interaction on the appropriate entity (again 1). DCC_{FA} is number of instantiations and destructions of sub-entities (depends on the level of the interaction). SC_{FD} is the number of actions to perform the checkpoint (total number of entities at the highest resolution) plus the number of actions to reflect the effects on an interaction on the appropriate entities (after translating a low-resolution interaction into many high-resolution interactions). DCC_{FD} is negligible and is counted merely by the initial creation and destruction of the set of entities. SC_{MRE} is the number of actions required to perform the checkpoint on a random level (depends on the number of entities at that level) plus the number of actions to reflect the effect of an interaction on the appropriate entity (always 1). DCC_{MRE} is the number of actions required to reflect an interaction down to all sub-entities and up to each parent (depends on the interaction level).

Figure 9 shows a plot of Simulation Cost (SC), Consistency Maintenance Cost (DCC) and Total Cost (SC+DCC) as measured in number of actions for the three schemes. For this plot,

- Number of resolution levels = L = 6
- Number of sub-entities = N = 8
- Total number of interactions = T = 10000
- Checkpoint Rate = R = 1/10
- Number of attributes = a = 7
- Interaction types = k = 5

The last two parameters do not affect our experiment, but are listed for completeness. We based the values of these parameters on likely values encountered in military simulations, making no claim about their universality. The actual values of the costs (measured in number of actions) is less important than the trends the curves display. Clearly, MRE incurs the least cost for this particular scenario. Experiments with other scenarios formed by varying the parameters listed above indicate similar trends.



Levels 6, FanOut 8, Total 10000, Rate 1/10

FIGURE 9: Cost comparison for FA, FD and MRE

Note that we allocated unit cost to both creation/ deletion of entities and traversing up/down the entity hierarchy. It may be that the latter is a much cheaper operation than the former. Accordingly, we measured the costs for the same scenario after making a change to allocate creation and deletion of an entity a cost that is an order of magnitude more than the cost of traversing the

1. We did not compute SCC for this experiment since it was a mock simulation requiring no behavioral design.

hierarchy. As expected, DCC_{FA} was the worst to suffer, as shown in Figure 10.

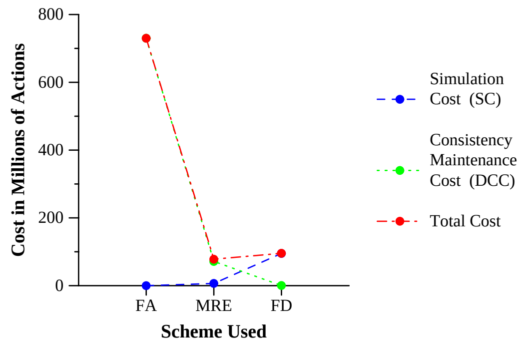


FIGURE 10: Cost comparison for FA, FD and MRE

5 Conclusions

The multi-resolution modeling problem poses significant challenges, particularly with respect to maintaining consistency among multiple concurrent levels of resolution of the same abstraction. Past attempts at this problem have been *ad hoc* in concept and in implementation.

The costs involved in designing a multi-resolution simulation are of crucial interest. On the one hand, a simulation that achieves consistency at a prohibitive cost may not be desirable. On the other hand, a low-cost simulation that compromises consistency is of little value. We expect the costs of implementing MREs based on ADGs will be overshadowed by the benefits of the resulting improved consistency. In addition, we believe that our approach may be able to achieve the right balance between simulation and consistency costs.

6 References

[Burd95] Burdick, C. D., Loral, *Interoperability of Simulations with Different Levels of Resolution*, DMSO Workshop, Washington, DC, November 29-30, 1995.

[Calder95] Calder, R. B., Peacock, J. C., Wise, B. P. Jr., Stanzione, T., Chamberlain, F., Panagos, J., *Implementation of a Dynamic Aggregation/ Deaggregation Process in the JPSD CLCGF*, Proceedings of the 5th Conference on Computer Generated Forces & Behavioral Representation, Orlando, Florida, May 1995.

[Dah95] Dahmann, J., Wood, D. C., *editors*, Special Issue of Proceedings of IEEE on Distributed Interactive Simulation, Volume 83, Number 8, August 1995.

[Davis93] Davis, P. K., Hillestad, R. J., *Families of Models that Cross Levels of Resolution: Issues for Design, Calibration and Management*, Proceedings of the 1993 Winter Simulation Conference, 1993.

[Hardy94] Hardy, D., Healy, M., *Constructive & Virtual Interoperation: A Technical Challenge*, Proceedings of the 4th Conference on Computer Generated Forces & Behavioral Representation, Orlando, Florida, 1994.

[Nat95] Natrajan, A., Nguyen-Tuong, A., *To disaggregate or not to disaggregate, that is not the question*, ELECSIM 95, Internet, April-June, 1995. `http://www.mystech.com/~smithr/elecsim95/PAPERS/natrajan/natrajan-main.html` or `ftp://ftp.cs.virginia.edu/pub/techreports/CS-95-18.ps.Z`

[Nat96] Natrajan, A., Reynolds, P. F. Jr., Srinivasan, S., *Consistency Maintenance using UNIFY* , University of Virginia CS Technical Report. `ftp://ftp.cs.virginia.edu/pub/techreports/CS-95-28.ps.Z`

[Reyn97] Reynolds, P. F. Jr., Srinivasan, S., Natrajan, A., *Consistency Maintenance in Multi-Resolution Simulations*, To appear in ACM TOMACS Special Issue on Training Simulations. `http://www.cs.virginia.edu/~an4m/tomacs.paper.ps.Z`

[Seidel95] Seidel, D. W., King, B. C., Burke, C. D., *AIM Approach to Simulation Interoperability*, The MITRE Corporation, Preliminary Draft, July 7, 1995.

[Smith95] Smith, R. D., *The Conflict Between Heterogeneous Simulation and Interoperability*, 17th Inter-Service/Industry Training, Simulation, and Education Conference (I/ITSEC) Proceedings, November 1995.

[Stober95] Stober, D. R., Kraus, M. K., Foss, W. F., Franceschini, R. W., Petty, M. D., *Survey of Constructive+Virtual Linkages*, Proceedings of the 5th Conference on Computer Generated Forces & Behavioral Representation, Orlando, Florida, May 1995.