Constructive Simulation with a Higher Automation Level for Training and Decision Support

Challenges of Today's Training and Decision Support

Typically, command and staff training requires multiple specific simulation systems for the creation of a realistic situation that fits to the training objective needed for the respective echelon level of the audience. The technical interface between those systems introduces higher cost if interaction between them is required, in a time where the demand for such interoperability is increasing. Especially, new emerging communication and technologies brings higher level of data and information exchange capabilities into the battlefield that crosses those classical boundaries. To reflect these capabilities within the training a merge or at least an interaction between previously separated simulations is required while facing the challenges of aggregation and disaggregation of data.

Also, the huge complexity of possible scenarios and the multi-variant evolvement of situations introduce at the same time multiple influencing variables into the decision-making process. Thus, a high realistic simulation environment is required which has the capabilities to depict a wide range of situational pictures reproducing real world behaviors with domain specific characteristics and details. The classical and simplest approach to achieve this level of realism is to use a huge personnel effort for manual leadership at low hierarchical levels and having only minimal automation. While having a realistic picture, at the same time reproducing the simulation is not possible due to the high involvement of human interaction.

Furthermore, the approach of the integration of man-in-the-loop with high personnel effort during simulation is not possible for the usage of such systems for the decision-making process. As a result, there is a long list of very specific simulation models answering domain specific questions (at high detail) quick but there is a lack of systems integrating and simulating the dependencies between them to obtain a constructive decision support at higher levels. Additionally there is always an information gap between the current situation and the ground truth. Introducing an additional step of simulation for decision support comes at the cost of additional time and often outdated data where the simulation result is based on. So, in the field of decision support tools, there is a demand for fast data exchange and maximum simulation speed presenting the feedback results in a way that they answer the questions directly without an additional complex analysis of simulation measures. Simply said: you need the results just in time to make the right decision.

Agent Based Simulation Model for Training and Decision Support

Even though, the training puts the focus on integrating the situation evolvement to improve the trainee's capabilities. The decision making is more about finding reasonable input based on simulation results to support the decision-making process. There is a way to bring these field together using the Agent Based Simulation approach and modern mathematical models. At the same time the computer processing power and network capabilities increased to support the huge amount of data processing required for the next step of higher automation and integration. Service oriented approaches and network distributed models for parallel computation open the door for more sophisticated simulation architectures solving these demands. A recent example is the combination of reinforcement learning and deep learning approaches in Google's DeepMind. First time, the computer was able to beat the best professional human players (Fan Hiu and Lee Sedol) in the Game Go, which is very complex regarding to possible variations and turns. This was considered as impossible to be solved by computers until then. The model is not able to cover all problems and questions for simulations right now, but this is just one example of what is possible using AI and simulation. This is what happened when the solution space changes and it is indeed continuously changing, be it infrastructure, mathematical models or others.

Now, Rheinmetall Electronics provides the constructive simulation system, named OSIRIS, which is capable for command and staff training on all levels within a single system. This is realized by using Agent Based Simulation Models combined with Markov Chain algorithms. The system especially raises the bar in the field of intelligent, automated behavior of units by enabling the control on higher echelon levels with autonomous execution of missions and tasks for a whole brigade, while simulating each single platform entity at the same time. The tasks autonomously executed cover all elements of land, air and sea. These include not only combat, but also combat support and combined arms activities as well as JOINT interactions. The high level of automation of OSIRIS give the opportunity to run scenarios without manual interaction as well as the possibility for control on low levels. If required, this system opens a wide range of applications in the field of Computer Assisted Training (CAX) and Decision Support System (DSS). Furthermore, OSIRIS provides the flexibility of changing doctrines and integrating or modifying the inventory of equipment and formations to assemble the agents and their capabilities, which is an option for Operations Research (OR).

The general theory and first agent based modelling approaches are going back to the early ‘70s idea of the segregation of dynamic models. At that time, there was not enough computing power to unleash the capabilities of the concepts and apply these to more sophisticated models. More recently, the advantage of new computing approaches and system architectures combined with Markov Chains-like algorithms allows to model even very large and complex scenarios with spread conflicts around the whole world. This advatange is realized within the OSIRIS simulation system. OSIRIS allows to model heterogenous behavior and the dependencies of other individuals explicitly by using separate agents. The following figure shows a simplified agent of OSIRIS running the contiguous OODA-like loop for the autonomous decision-making process. The process of the agent considers the assignment of the current task including related restrictions (like combat boundaries, conditional time frames, priorities, …) to achieve the goal using a heuristic solving strategy. The strategy is based on a heuristic model (using the built-in agents). After that the Agent acts autonomously according to the found solution. A manual interaction can interfere the acting by changing or refining the assigned task (e.g. to override a behavior or specialize to achieve a specific result).

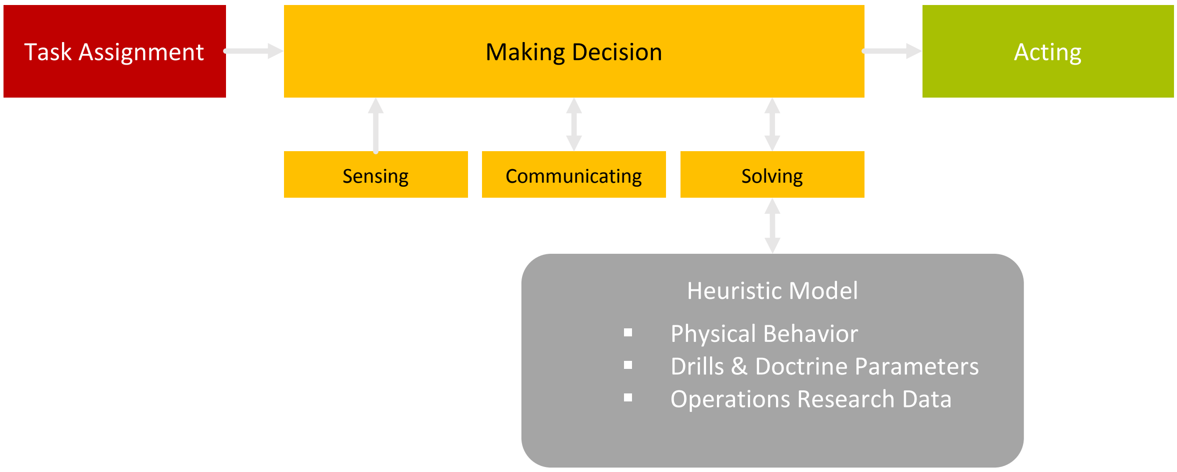


Figure 1: Simplified picture of a single Agent of OSIRIS using heuristic model for solving strategy

The acting of an agent can include the assignment of tasks to other subordinate agents (or requests to hierarchically higher levels). The following figure shows the hierarchy of assigning tasks to subordinate agents based upon the solving result.

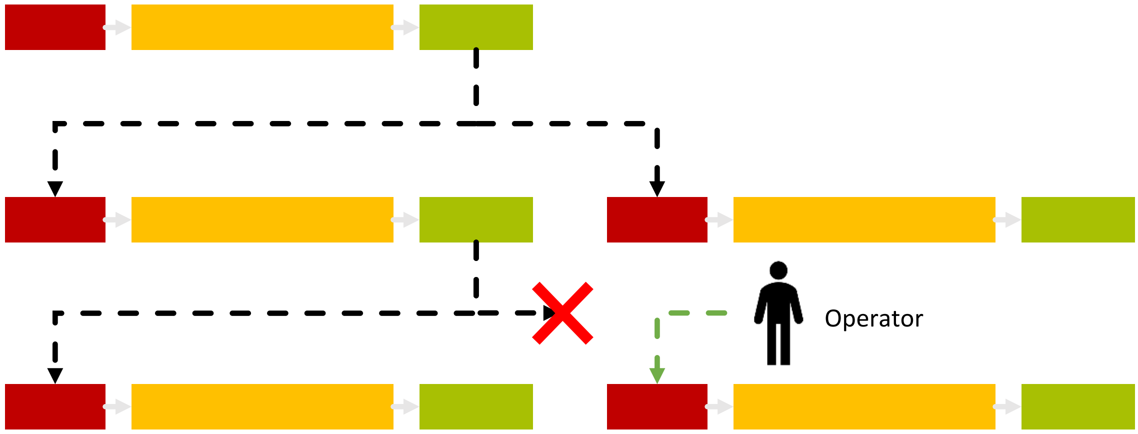


Figure 2: Simplified picture of Agent hierarchy with manual interaction in OSIRIS

Agents can be applied to different levels in the hierarchy and can be assembled to reflect structures like a military ORBAT. The manual interaction as seen in the picture where an operator gives a specific task, results also in an autonomously reorganization of the rest of the task groups (e.g. apply changes to other subordinate tasks according to the changed environment, e.g. less available troops to achieve the goal).

One salient point of the Agent Based Model of OSIRIS is the capability to have single platforms with real unique identities instead of just having an aggregated value like most constructive simulations for higher echelon levels (e.g. statistical accumulation of number of tank and resulting capabilities). This results in resolving the problem of aggregation and disaggregation.

Combining the Agent Based Simulation Model with Markov Chains

The concept of Markov Chain's itself is going back to 1913, where the Russian mathematician Andrey Markov a model of chains used to model the alliteration of vowels and consonants. A discrete time stochastic control process was derived as a mathematical general-purpose tool – the Markov Decision Process (MDP) allowing flexibly applying this to other fields. The Markov Chain can be used to describe the probability of transitioning from one state to any other possible state. Most importantly there is no history about the previous states required to express the possible future, which means the probability of an upcoming state only depends upon the its previous state. This makes the model extremely performant in computation as well as expressive with flexible integration of variants at the same time. The following figure shows a simple chain having two possible states and probabilities to switch between states (or next state is same state). This is a very simplified view onto the topic, because the Hidden Markov Model (HMM) as used in OSIRIS within the heuristic solving is naturally much more complex.

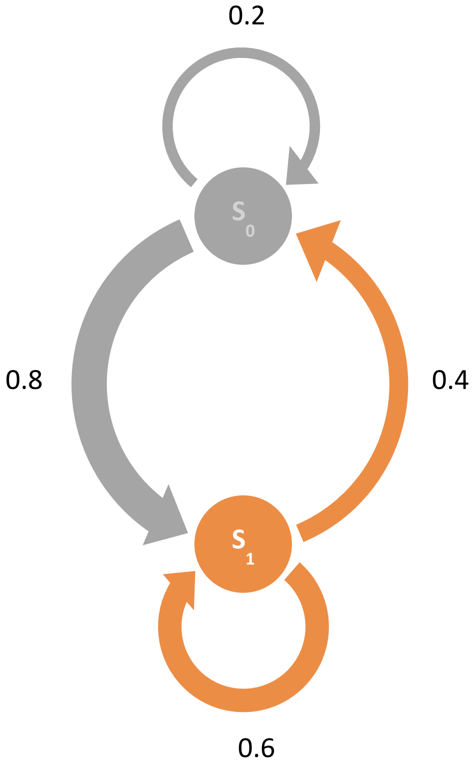


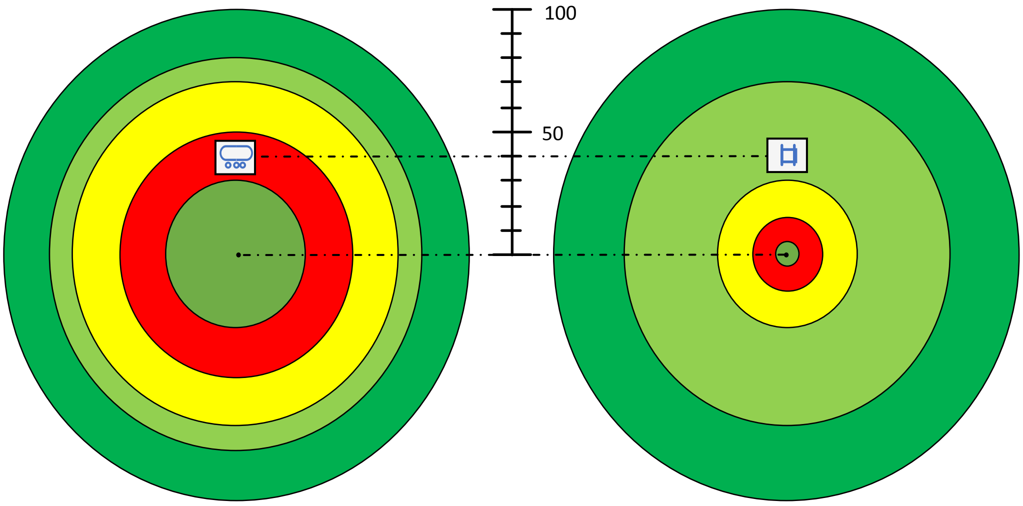
Figure 3: Example of simple Markov Chain expressing to states

OSIRIS is highly flexible because of its Agent Based Model combined with Markov Chains. Due to the agent based approach parts of the model can be also used in combination with other solving strategies, if required.

Practical Usage Examples within OSIRIS

Example 1: Damage Calculation

A practical usage example of the previously described and realized concepts within OSIRIS is the probability expression of damage calculation as shown in the following figure. The constructive simulation system OSIRIS calculates every single shot of a platform (including type of ammunition used, etc.) and how this affects the target. In the example below, when ammunition with fragmentation and effect within a certain range is used, the damage within the range of the hit point is calculated. So even if not directly hit, damage is calculated according to the model and the probability distribution in relation to ammunition capabilities and protection level of the target platform. Naturally, the damage effect is minimized when the potential target is covered (e.g. using revetment or installations).



Target Damage Level		None	Light	Medium	Heavy	Lost
Target Protection Level	Light	0,2	0,1	0,2	0,2	0,3
	Heavy	0,3	0,4	0,15	0,1	0,05

Ammunition:	155 mm Art
Fragmentation Radius	100 m
Target Type	APC

Figure 4: Damage calculation example

Example 2: Visualization of the Battlefield and Integration with Virtual Simulation

As previously said, the modern warfare brings new information technology on the field. Therefore, the training environment must be able to adopt those features as well to deliver a high realistic experience using new tactical elements. An example are Live-Video streams from drones. High resolution of OSIRIS having unique platforms with specific attributes (like position, status) simplifies the projection of a high detailed view using an Image Generator (IG) as used for virtual simulations. Due to the Agent Based Model, where each single Agent can be handled specifically, the complete handover of a single or multiple agents to another simulation is possible. As an example, this allows OSIRIS providing very realistic footage generated out of the scenario (instead of scripted film snippets) to the training audience. Also, the control for special tasks and roles can be integrated.