2009

Conceptual Requirements for Command and Control Languages

Andreas Tolk
Old Dominion University, atolk@odu.edu

Curtis L. Blais

Saikou Y. Diallo
Old Dominion University, sdiallo@odu.edu

Charles Turnitsa
Old Dominion University, cturnits@odu.edu

Original Publication Citation

Tolk, A., Blais, C. L., Diallo, S. Y., & Turnitsa, C. (2009). *Conceptual requirements for command and control languages.* Paper presented at the 2009 Fall Simulation Interoperability Workshop, Orlando, FL.

Conceptual Requirements for Command and Control Languages

Andreas Tolk, Ph.D.
Frank Batten College of
Engineering & Technology
Old Dominion University
Norfolk, VA 23529
atolk@odu.edu

Curtis L. Blais
MOVES Institute
Naval Postgraduate School
Monterey, CA 93943
clblais@nps.edu

Saikou Y. Diallo, Charles Turnitsa
Virginia Modeling Analysis and
Simulation Center
Old Dominion University
Suffolk, VA 23435
[sdiallo, cturnits]@odu.edu

Keywords:

Coalition Battle Management Language (C-BML), Command and Control Languages,
Community-of-Interest (COI) Languages, Conceptual Modeling

ABSTRACT: *The current Coalition Battle Management Language initiative (C-BML) will define a language to unambiguously exchange command and control information between systems. This paper introduces a categorization that may be used to guide the process of developing C-BML effectively by enumerating the conceptual requirements the authors have identified in model-based data engineering and process engineering based studies in various domains.*

First, it is important to distinguish if application of the language will support the planning, execution, or observation phase of command and control. While C-BML already distinguishes between tasking and reporting, planning is a category with different requirements.

Second, the language must be able to express various spatio-temporal constraints, which can be expressed using fixed expressions, relative to each other, or in mixed forms. In addition to the traditional spatio-temporal constraints, operation-specific constraints – or the perception thereof – need to be expressed.

Finally, it must be determined if the constraints are used in support of accomplishment-driven objectives or avoidance-driven objectives. While this category seems to be trivial to most human consumers of the language, it has significant implications for systems.

The paper introduces the conceptual constraints using examples and evaluates mathematical means provided by discrete structures needed for computation to describe their ability to cope with these challenges.

1 Introduction

The Simulation Interoperability Standards Organization (SISO) supports currently the standardization efforts to create a Coalition Battle Management Language (C-BML). C-BML *is defined as the unambiguous language used to command and control forces and equipment conducting military operations and to provide for situational awareness and a shared, common operational picture.* [1]

According to Boyd, decision-making occurs in a recurring cycle of observe-orient-decide-act [2]. BML contributes to this cycle by providing reports that communicate the results of the observation phase for orientation and by tasks to communicate the decision and make it actionable. As such, BML is a means of communication and not part of the cognitive process. BML is used to communicate results: reports are the result of observation (in the general sense), and tasks are the result of the decision process. In addition, plans need to be shared, and hence communicated, as well. The traditional schema shared worldwide in the armed forces is referred to as the “5Ws:” WHO is doing WHAT WHERE WHEN and WHY. WHO refers to an actor, WHAT refers to an action or activity, WHERE refers to a location, WHEN refers to a time, and WHY refers to the underlying motivation or the intention. This general schema is applicable to tasking as well as for supporting reporting.

However, while this general schema holds in many applications and domains, it needs to be extended to be unambiguous for applications. A simple example is the WHO-WHAT-WHERE relation. To be applicable, it needs to be specified if the location indicates where the actor is, where the action takes place, or if both locations are – or need to be – provided. Another example is the specification of the action WHAT. If the addressee is capable of planning and has resources of its own, the WHAT can be a simple task reference, which means a task is given by just using a term and no details on how to conduct the task. This still is well known as *"Auftragstaktik:"* the taskee decides how to achieve the task objective within the current constraints. However, if the addressee is a robot or a machine without planning intelligence, it may be necessary to specify the action in much more detail, like breaking every action down into directly executable tasks for individual entities that are the organizational part of the WHO. BML must be able to accommodate all these cases.

Hieb and Schade published several papers on necessary additions, extensions, and enhancements of this simple schema of the 5Ws for reporting and tasking, including the representation of intent. The complete grammar for BML that evolved from these ideas is called Command and Control Lexical Grammar (C2LG) [3]. Their extensions are also driven by the need for a more context-specific specification, such as introducing the need to include the tasking organization in orders, the observing organization in reports, etc.

In this context, this paper enumerates several conceptual requirements that have to be supported by command and control (C2) languages in general and by C-BML in particular. It starts with the evaluation of communication concepts needed in support of planning for an operation, tasking of organizations, and observations. The next section will deal with constraint definitions addressing space and time (spatio-temporal) as well as parallel ongoing operations that affect the execution of the addressed task. Finally, the intention may be to accomplish an envisioned state or to avoid an envisioned state. The later is often connected with denying the opponent the ability to reach his objectives. The paper seeks to establish a research frame and contribute to a requirement catalog for successful C2 languages, including C-BML.

2 Supporting the Planning, Tasking, and Observation Phases

C2 languages must support all phases of the military decision process. It should be pointed out that in the context of this paper the emphasis is on the resulting information exchange, not the way this information is presented. We will focus on the three phases of planning, tasking, and observing. We will use the idea of the 5Ws to demonstrate the various concepts that need to be supported and to introduce the idea of a decision matrix in support of interoperation.

2.1 Supporting the Planning Phase

The 5Ws answer the question of WHO is doing WHAT WHERE WHEN and WHY. The main concept that needs to be supported in the planning phase is to distinguish between WHO-types and WHO-items; in other words: in the planning phase, it is not necessary to know the exact unit (WHO-item) that is going to conduct the WHAT, it is just necessary to know that certain unit types (WHO-types) have the required capabilities enabling them to conduct the WHAT.

This requires modeling the capabilities required for activities independently from the units, and even the unit types. It is recommended to model capabilities as properties of system types in context (and this context can be empty, which means that a system of this type can apply the respective capability in every context). The context can be defined using all concepts in this paper, such as spatio-temporal constraints, operational constraints, and using measures of merit based on accomplishment or avoidance.

At the end of the planning phase, the currently available units are compared regarding their available systems and current contexts – resulting in applicable capabilities of the unit – with the required capabilities in order to achieve the objective.

A C2 language must be able to express the type-specific capabilities including the constraints and contexts independent from concrete instantiations. It must be possible, for example, to talk about the ability to attack a hostile unit – or unit type – and change the state – or potential state – to make this unit no longer usable for hostile operations.

In other words, the language must support the description of capabilities in spatio-temporal and operational contexts in support of selecting the best instantiations at the end of the planning phase. This shows that selection, scheduling, and orchestration become subsumed within the spatio-temporal and operational contexts that define the capabilities of the WHO-type, which is at the center of the planning phase along with the WHAT under WHERE/WHEN constraints.

For interoperating systems, the planning phase specifies the need to execute a concept of operations across multiple simulations. As each simulator implements this concept of operation according to a deterministic state machine, there are different state/input/output pairs for each planned action depending on the context, the doctrine, and the rules of engagement. While federation developers should be aware of the variance between executions and adjust the federation accordingly, C-BML should not carry the responsibility of specifying the state machine that a system must use in the execution of a task. During the analysis process, however, it is important to examine the results not based on the C-BML messages exchanged between systems and how they were intended to be executed but rather on how they are actually implemented in the receiving system's state machine. Conceptually, this observation means that the federation of systems using C-BML is necessarily incomplete in that it does not and cannot mandate how a plan is implemented in a system. C-BML should, however, provide a decision matrix that is mapable to the executing system's state machine.

A decision matrix is a matrix indicating which task(s) to perform given an event. The decision matrix captures not only the rules of engagement but also the relevant doctrine. Currently rules of engagement and doctrine are captured in free text form in the C2 world and implemented as state machines in an M&S system. The decision matrix bridges the two worlds by allowing multiple state machines to be generated from a given matrix.

Table 1: Sample Decision Matrix for a Concept of Operation

⮣	Process1	Process2	Process3	Process4
Process5	Event6	Event4	Event9	Event27
Process2	Event8	Event5	Event10	Event54
Process3	Event3	Event7	Event12	Event43
Process8	Event12	Event6	Event46	Event87

Table 1 shows a sample of an imaginary plan in which tasks are modeled as processes and events (call for fire, securing a building). Additional information is pulled from the rules of engagement to address the eventuality of certain events (return fire when fired upon) and the required response in the form of a task that has to be performed. The arrow shows the direction of the matrix and reads "when in Process *i* and event *j* occurs perform process *K*". There is at least one task per process and when there is more than one task, we refer to the process as a composite task or order for functionally related tasks. It is also worth noting that a state is a snapshot of a WHO-item(s) within a process. An event can occur anytime within the execution of a process. For instance the matrix can capture the following: "when responding to a call for fire and another call is received, verify the priority of fire before proceeding and respond to the one with the highest priority." In this case "responding to a call for fire" is one process that includes multiple tasks and the reception of another call for fire is an event that triggers the process of adjudicating the priority of fire.

The matrix generated from C2 systems can be used to generate state diagrams that can either be used to identify a suitable M&S system or validate an already chosen system. This validation process consists mainly of comparing the transitions within the state machine derived from the matrix with that of the M&S system. C-BML can refer to a given matrix for the execution of certain tasks if the planner deems it important. The tasking phase will consist of assigning existing WHO-items to the tasks identified in the planning phase and choreographing the execution to match the concept of operation with the constraints defined in the decision matrix.

In summary, the planning phase is concerned with general abilities as normally captured by types of actions, processes, and entities that are in principle able conduct them. For short term planning, the abilities available in the sphere of influence may have to be taken into account as well. Planning applications need therefore the ability to communicate general and actual or instantiated abilities for the conducted as well as the targeted side regarding general and actual and instantiated properties of actions and processes.

2.2 Supporting the Tasking Phase

For the tasking phase it is important to unambiguously define constraints and objectives for each task. Both categories will be discussed in sections to follow.

If the planning phase is done correctly, tasking can directly evolve from planning by selecting WHO-items to conduct the planned operations.

Constraining tasks require the handling of "negative tasks" as in "do not cross phase line alpha." Especially in machine-to-machine interoperation, constraints must be expressed specifically as tasks in order to be handled. Another aspect of constraints is the notion of decision structure. In general, the decision structure or decision matrix is implicit as it follows established doctrine. For machine-to-machine interoperation, however, tasks that involve simultaneous or quasi-simultaneous events (call for fire for example) have to be handled by the decision matrix. The language should be able to specify which decision matrix to be use if and when required and even specify conditions under which a given matrix is usable. Additional aspects of tasking, such as functional and temporal dependencies between tasks and starting and ending conditions, should not only derive from the planning phase but also from the observing phase as the operation unfolds.

Another related aspect is the specification of rules of engagement which determine the behavior of entities in a given situation. In terms of interoperation, it cannot be assumed that all systems exchanging information are in identical situations or have the same rules of engagement for the same situation. Consequently it is important to determine whether it is the responsibility of the tasking system to specify which rules to use or if it is left to the executing system to behave according to its own rules. This decision is equivalent to selecting an appropriate decision matrix as discussed earlier.

In summary, the applications supporting the tasking phase need to communicate instantiated abilities and constraints. If planning is merged into tasking, this should be doable be assigning instantiating objects – entities, actions, and processes – to the types of the planning phase. It needs to be assured that the available ability covers the required ability. It must also be allowed that objects that expose the needed ability can be assigned even if their type does not necessarily expose this ability. An example is a personnel intense artillery unit conducting police operations.

2.3 Supporting the Observing Phase

The observing phase results in reports about own, opposing and neutral forces and actors, again following the 5Ws. However, the observations may not always result in the necessary data needed for unambiguous population of the 5Ws. Nonetheless, this information needs to be reported. While the information itself may be ambiguous, the representation must be unambiguous. Examples for ambiguous information comprise:

- Incompleteness (not all pieces are available)
- Contradictions (two mutually exclusive reports on the same object)
- Uncertainty (only the likelihood of alternatives is known)
- Vagueness (missing accuracy)

Several mathematical concepts have been developed to deal with representation of ambiguity [4]. It is recommended to include them in the specification, as they are applicable to all spatio-temporal observations. More recent work on ontological means to capture uncertain information, as summarized in [5], should become part of future phases, as they allow the mediation between different representations as they have to be expected when supporting heterogeneous C2 and M&S systems.

The challenge of representing uncertainty for a machine or a system is that the ambiguous information that is real must be communicated and presented in an unambiguous form. The message "approximately 10 to 12 soldiers, likely hostile, have been seen near the bridgehead" has multiple interpretations. This challenge is known since the early days of machine-based knowledge representation and has not been solved so far. Nonetheless, for supporting the observing phase, the challenges must at least be captured in machine understandable form.

3 Spatio-temporal and Operational Constraints

The Command and Control process results in orders that are normally constrained by pre-conditions and post-conditions. This principle ripples through to the executable tasks for taskable units. If these constraints are ignored, the results will not reflect the intent of the command. For example, it makes no sense for a unit to start a major attack operation at the wrong place (spatial constraint) or at the wrong time (temporal constraint). Even more complicated are operational constraints, such as

- What is the situation of my neighbors?
- What is the situation of my combat support?
- What is my own logistical situation?
- Is the overall situation progressing as planned?
- And more.

3.1 Temporal Considerations

The spatio-temporal constraints have been researched in great detail in the Geographic Information Systems (GIS) community, as among others described in [6]. Temporal constraints are well known in the C2 as well as the M&S community. A summary of possibilities is given in [7]. The Time calculus published by Allen [8] is still used, and even is reflected one-to-one in the Joint Consultation Command and Control Information Exchange Data Model (JC3IEDM). Allen defined seven relations allowing computing the order of related events:

- X **before** Y (X ends before Y starts)
- X **meets** Y (Y starts when X ends)
- X **overlaps** Y (X starts before Y ends)
- X **during** Y (Y starts before X starts and Y ends before X ends)
- X **starts** Y (X and Y start at the same time)
- X **finishes** Y (X and Y end at the same time)
- X **equal** Y (x and Y start and end at the same time)

In [9], these ideas are generalized to analyze temporal relations within language constructs, which is of help for C2 languages as well, in particular when such logical expressions shall be extracted from written communications, such as manuals or operational orders. These concepts also allow intelligent systems, such as agents, to reason over temporal constraints.

In addition to this qualitative temporal concepts, quantitative concepts using crisp points in times or time intervals need to be supported.

3.2 Spatial Considerations

The need to align the concepts, as used in GIS and C2 systems, has been articulated already in [10]. Results of research on this topic have only been sparsely published, so far. One the technical challenges is that most C2 systems applications are coordination focused, while GIS applications use vector data. However, such problems can be overcome, as the preferences are translatable into each other. What is needed regarding the conceptual requirements for C2 languages is the definition of points, lines, areas, and spheres and their spatial relation to each other. It should be pointed out that spatial constraints can define where an operation should take place as well as exclude certain areas as well. All these spatial constraints can have temporal or operational constraints attached as well, such as coordination lines and other control features are only valid between certain points in time, or air coordination means are only needed as long as own aircrafts are available.

Similar to the work that Allen has performed in the area of a temporal algebra, [11] has enumerated a similar algebra for spatial relationships. This has been expanded, and formalized, with regards to point-set topological reference systems, as well as adjacent region reference systems [11], which should allow a merged algebra to deal with the problems of aligning the GIS view of vector based models with C2 and M&S models that are based on a point-set representation. The relationships identified are (for two regions, A and B):

- A is **disconnected** from B
- A is **part** of B
- A is a **proper part** of B
- A is an **equivalent coincident** of B
- A **overlaps** B
- A **partially overlaps** B
- A **externally connects** with B
- A is a **proper connected part** of B
- A is a **proper non-connected part** of B

Axioms defining these relationships are part of Region Connection Calculus theory (RCC), having been developed for some time at the University of Leeds [12, 13]. This field of research, a subset of graph theory, is known as mereology, and will be an important part of an ontological representation of actors and actions in a battlespace.

3.3 Operational Considerations

Operational constraints are the most demanding ones, as they require the languages to capture in logic, i.e. in a machine understandable way, to capture the success or progress of an operation. Similar to measuring success of an overall operations, a function with thresholds needs to be defined that is used for a machine to trigger the decisions. As mentioned before, constraints like “the logistical situation is sufficient to enable the attack” must be captured, which needs to be translated in “enough fuel and ammunition is for the current operation available.” However, how much fuel depends on the terrain, the weather, and the category of operation, the amount of ammunition needed depends on the enemy, the education of the own soldiers, etc. In other words, the metrics must be adaptable to the situation allowing situation decision to avoid structural variances, as introduced to SISO in [14].

Most likely, due to the operational demands of the domain, with C2 systems (also including M&S and decision support systems) many tempo-spatial relationships will be expressed with relative (rather than fixed) values in a relationship. This is true in all uses of such systems (planning, training, operations, and analysis), and as pointed out earlier the phases of such uses (planning, tasking, and observing) may include such reference with certainty and precision, or may be expressed with some certainty blurring quality (stochastic probabilities, uncertainty, ‘fuzziness’, and others).

In his dissertation [15], Schnurer introduced a multitude of geospatial operators needed for machines to understand tactical situations, such as a break-through, an open flank, or the sufficient distance to allow for artillery attacks. Such constructs are needed in addition to simple definitions of terms, as the unambiguous definition of terms must include the unambiguous representation for machines in the form of logic as well. As the geospatial representation in different simulation and C2 systems is heterogeneous, this adds another level of complexity to the required supporting functionality.

3.4 The Way forward: A Tempo-Spatial Algebra

With regards to possible uses of such information, and the possible phases of representing such information, the topic of spatio-temporal references for a C2 language are certainly an area requiring more research. However, before exploring such a claim, it is necessary to realize that the fact that any such references may have to capture not only concrete values, but also relative values shows that not only a series of reference systems are needed, but a richer method such as a symbolic algebra, that will allow for the representation of either spatial or temporal values in symbolic terms.

It was mentioned above that the study of regional overlaps, or parts of a whole, is known as mereology. If temporal considerations are viewed as part of a graph representation of time, then a similar symbolic algebra should be possible to relate both the spatial dimensions to the temporal dimensions, in a unified language that ties both together. This unified algebra will have a higher dimensionality than either separate domain requires, but will allow a cross domain representation of each in the terms of the other. This becomes important, when dealing with not only relative values for time or space, but more importantly when dealing with such values that change dynamically (as they do in any operation). The work in [5, 6, 8, 12, 13] will prove invaluable in deriving such an algebra, yet the ability to use it in concert with a C2 language, or grammar must be remembered throughout the research process.

In summary, the "what-when" combinations are much more complex than it has been addressed so far. One of the most challenging aspects will be to capture and communicate tactical situations on the battlefield in machine understandable form. This task is subject of ongoing research, as it is closely related to the task to support machine-based situational awareness as well, as the same functionality is needed to support cognitive processes based on spot-reports and snap-shots of situations, as provided by common operational pictures.

4 Metrics and Measures of Merit: Accomplishment and Avoidance Driven Objectives

Metrics are not only needed to measure the success of an operation, they are also needed to measure thresholds of constraints for operations or tasks. The objective of an operation falls normally in one of two categories: the task is conducted to accomplish something (like building a bridge, securing an area, reaching a certain point at a given time, etc.) or to avoid something (like denying enemy access to certain resources, etc.)

C2 languages must support both categories to define metrics. In addition, mixed forms must be expressible. The task "march from point A to point B avoiding area C" has both metrics combined, as the accomplishment-driven part is to reach point B and the avoidance-driven part is to avoid area C.

How to combine accomplishment and avoidance driven objectives into metrics for tactical and operational support of operations and apply them in utility functions is documented in [16]. During the underlying experiments, more than 70,000 simulation runs had to be evaluated. The success of these operations was determined by the number of disabled hostile units as well as by minimizing the number of hostile units successfully breaking through a line of defense. The approach was presented to NATO in more detail in [17]. Similar to the discussion on operational constraints, it is not sufficient to define the terms for the metrics, but the formula to be applied needs to be communicated as well to ensure unambiguous communications between systems. If one system bases the definition of a successful breakthrough battle on remaining forces in the objective area, but the other system defines the success using the resulting combat power ratio at the end of the battle, using of well defined terms is not sufficient. The language must therefore be bale to communicate measures of success, and these metrics must be defined by soldiers, but understandable by soldiers and machine.

In summary, metrics must be based on operational warfighter definitions – not model artifacts – and be communicated in machine-understandable form.

5 Summary

The conceptual requirements for C2 languages require agreeing on concepts representing not only tasks, tasker, and taskees for the traditional 5 W: "Who is doing What, Where, When, and Why," but spatio-temporal and operational constraints with enabling metrics are needed as well. These concepts then need to be composed based on construction mechanisms, such as grammars, production rules, or other adequate mathematical tools, as covered in [16], into sentences – or regular expressions – of the C2 language.

While the construction mechanism is important to support parsers, the focus of conceptual work should lie on the underlying conceptual model, as only common conceptualization enable the lossless mediation between viewpoints represented by alternative implementations.

Acknowledgments

The underlying research was partly supported by the Joint Advanced Training Technology Laboratory (JATTL) of the US Joint Forces Command, the US Army Test and Evaluation Command (ATEC), and NATO's Research and Technology Organization (RTO). We also like recognize the valuable feedback received from colleagues of the US Department of Homeland Security and the US Department of Energy.

6 References

[1] Sudnikovich, W.P., J.M. Pullen, M.S. Kleiner, and S.A. Carey (2004). Extensible Battle Management Language as a Transformation Enabler. *SIMULATION*, 80:669-680

[2] Defense and the National Interest (DNI) (2007). John Boyd Compendium: A Discourse on Winning and Losing. Online accessible via http://www.d-n-i.net/dni/john-r-boyd/

[3] Schade, U., and M.R. Hieb (2007). Improving Planning and Replanning: Using a Formal Grammar to Automate Processing of Command and Control Information for Decision Support. *International C2 Journal*, 1 (2): 69-90

[4] Kruse, R., E. Schwecke, and J. Heinsohn (1991) *Uncertainty and Vagueness in Knowledge Based Systems: Numerical Methods*, Springer

[5] Laskey, K.B., and P.C.G. Costa (2009) Uncertainty Representation and Reasoning in Complex Systems. In Tolk, A., and L.C. Jain (eds.) *Complex Systems in Knowledge-Based Environments: Theory, Models, and Applications*, SCI 168, Springer, pp. 7-40

[6] Parent, C., S. Spaccapietra, and E. Zimányi (1999) Spatio-temporal conceptual models: data structures and space and time. In *Proceedings of the 7th ACM international Symposium on Advances in Geographic information Systems* (GIS '99). ACM, New York, NY, 26-33

[7] Ferro, L., I. Mani, B. Sundheim, and G. Wilson (2000) Tides temporal annotation guidelines. Technical report, The MITRE Corporation.

[8] Allen, J. (1983) Maintaining Knowledge about Temporal Intervals, *CACM* 26 (11): 832-843

[9] Lapata M., and A. Lascarides (2006) Learning Sentence-internal Temporal Relations. *Journal of Artificial Intelligence Research*, 27: 85-117

[10] Powers, M., M.R. Hieb, J.M. Pullen, and M. Kleiner (2006) A Geospatial Battle Management Language (GeoBML) for Terrain Reasoning. In *Proceedings of the 11th International Command and Control Research and Technology Symposium (ICCRTS)*, Cambridge, UK

[11] Randell, D.A., Cui, Z., and Cohn, A.G. (1992) A spatial logic based on regions and logic. Proceedings 3rd Int. Conf. on Knowledge Representation and Reasoning, San Mateo, pp. 165-176

[12] Cohn, A.G., and Gooday, J.M. (1994) Defining the Syntax and Semantics of a Visual Programming Language in a Spatial Logic. In Anger, F.D., and Loganantharaj, R. (eds.) Proceedings of AAAI-94 Spatial and Temporal Reasoning Workshop

[13] Gotts, N.M., Gooday, J.M., and Cohn, A.G. (1995) A Connection Based Approach to Commonsense Topological Description and Reasoning. Division of Artificial Intelligence, School of Computer Science, Leeds University, Leeds, UK

[14] Tolk, A. (1999) Non-Monotonicities in HLA-Federations. In *Proceedings of the Spring Simulation Interoperability Workshop*, Orlando, FL: pp. 1-7

[15] Schnurer, R. (1996). Zur Abbildung von Fuehrungsprozessen in geschlossenen Gefechts-simulationsmodellen (transl.: *On Modeling of Command Processes in Closed Combat Simulation Models*). PhD Thesis at the University of the Federal Armed Forces, Pro Universitate Verlag, Sinzheim, Germany

[16] Hofmann, H.W., R. Schnurer, A. Tolk (1994). KOSMOS Simulation Experiments on Stable Defense. Appendix 3 to Annex IV to AC/243 (Panel 7) TR/5 - Technical Report NATO Research Group 18, Brussels

[17] Schnurer, R., and A. Tolk (1995) Using Fuzzy Sets within an Automatic Assessment Procedure for the Evaluation of Stochastic Simulation Results. Proceedings of the NATO AC/243 (Panel 7) Symposium on Uncertainty, January, Shape Technical Center, The Hague

Authors' Biographies

ANDREAS TOLK is Associate Professor in Engineering Management and Systems Engineering at Old Dominion University (ODU) in Norfolk, Virginia. He is affiliated with the Virginia Modeling Analysis & Simulation Center (VMASC). His domain of expertise is the integration of M&S functionality into real world applications based on open standards. He received a Ph.D. and an M.S in Computer Science from the University of the Federal Armed Forces in Munich, Germany.

CURTIS BLAIS is a Research Associate in the Naval Postgraduate School Modeling, Virtual Environments, and Simulation (MOVES) Institute. His research interests include modeling irregular warfare, development of metadata describing modeling and simulation resources, and application of Semantic Web technologies to improve interoperability across C2 and M&S systems. He received a B.S. and M.S. degrees in Mathematics at the University of Notre Dame and is currently working on a Ph.D. in MOVES.

SAIKOU Y. DIALLO is a Ph.D. candidate and Senior Project Scientist at the Virginia Modeling Analysis and Simulation Center (VMASC) of the Old Dominion University (ODU). He received his B.S. in Computer Engineering (2003) and his M.S in Modeling & Simulation (2006) from ODU. His Ph.D. research focuses on the domain of Model Based Data Engineering and Web Services for M&S applications.

CHARLES TURNITSA is a Senior Project Scientist at the Virginia Modeling Analysis and Simulation Center at Old Dominion University. In addition he is also a Ph.D. Candidate, studying under Dr. Andreas Tolk at ODU. His Ph.D. research is in the area of formalizing a standard representation of processes for Modeling and Simulation. He has a M.S. in Electrical and Computer Engineering from that institution.