

Research Article

Communication Analysis of Network-Centric Warfare via Transformation of System of Systems Model into Integrated System Model Using Neural Network

Bong Gu Kang ¹, Kyung-Min Seo ², and Tag Gon Kim ¹

¹Department of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Republic of Korea

²Naval & Energy System R&D Institute, Daewoo Shipbuilding & Marine Engineering (DSME) Co., Ltd., Seoul, Republic of Korea

Correspondence should be addressed to Kyung-Min Seo; kmseo.kumsung@gmail.com

Received 11 December 2017; Accepted 11 March 2018; Published 24 June 2018

Academic Editor: Arturo Buscarino

Copyright © 2018 Bong Gu Kang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Communication system in the network-centric warfare (NCW) has been analyzed from the perspective of the system of systems (SoS), which consists of a combat system and a network system so that the two reflect each other's effects. However, this paradoxically causes a prolonged execution time. To solve this problem, this paper proposes an advanced integrated modeling method for the communication analysis in the NCW via the transformation of the SoS, which reduces the simulation execution time while ensuring the accuracy of the communication effects. The proposed models mainly cover interentity traffic and intraentity mobility developed in the form of feed-forward neural networks to guarantee two-way interactions between the combat system and the network system. Because they are characterized as discrete events, the proposed models are designed with the discrete-event system specification (DEVS) formalism. The experimental results show that the proposed transformation reduced an error by 6.40% compared to the existing method and reduced the execution time 3.78-fold compared to the SoS-based NCW simulation.

1. Introduction

Network-centric warfare (NCW) is an emerging theory of war based on the concepts of nonlinearity and complexity [1, 2]. According to the doctrine of NCW, all combat entities, such as war-fighters, manned and unmanned platforms, and command and control (C2) centers, are linked via a network architecture for sharing the combat situation [3, 4]. For example, network nodes are dynamically located as the entities are moved; traffic between the nodes is also changed whenever they interact in joint operations. Thus, NCW should be analyzed based on the communication interactions, which mainly cover interentity traffic and intraentity mobility [5, 6].

For communication analysis in NCW, modeling and simulation (M&S) techniques have been widely used; and a well-known approach is to simulate communication factors separately within the whole NCW simulation [7, 8]. In other words, an M&S engineer separates a network system for explicit communication information from a combat system that generates the overall combat scenario. Then, the

engineer interacts with the two systems under standardized architectures such as high-level architecture (HLA) or test and training enabling architecture [9, 10]. This is a typical approach for system of systems (SoS) development, which supports the high reality of component systems and flexibility in changing them [11, 12].

Despite these advantages, the SoS-based simulation causes a prolonged simulation execution time. Simulation analysis generally requires performing simulation evaluations of all possible input combinations as a “what if” analysis; thus, it consumes a lot of execution time due to the many repeated experiments [13]. In the SoS approach, the simulation overhead due to the use of the interoperation architecture is a primary reason for the time-consuming problem [14, 15].

To overcome this weakness, this paper focuses on a transformation approach that shifts from an SoS-based NCW simulation to an integrated form without using the interoperation infrastructure. For simulation analysis of the network system, we distinguish two main factors: interentity traffic

and intraentity mobility. Based on the factors, we propose (1) a new combat model that includes them and we describe (2) how the model is connected to the network model in the manner of an integrated simulation.

Traffic and mobility, in the integrated simulation, should be two-way interactive with the combat model and the network model. For example, the combat model requests the traffic situation, such as packet delivery ratio and delay, between the source and the destination nodes to the network model. Then, the network model eventually responds to the request based on the network conditions. Thus, the combat model proposed in this paper facilitates the representation of interactive changeability according to the current state of the network; and the network model enables an analysis of the communication effect more realistically. This interactive design for the two models represents a clear difference from previous studies on the development of traffic and mobility models.

To this end, this paper proposes a transformation process consisting of the following three major phases: (1) traffic and mobility data acquisition from SoS-based simulation, (2) data preprocessing for training, and (3) traffic and mobility models hypothesis and variable estimation of them using a neural network. In the third phase, we design regression models in the form of feed-forward neural networks. Specifically, inputs of the models are the communication effects, for example, packet delivery ratio (PDR) and end-to-end delay, and outputs are the variables, for example, interdeparture time of traffic generation and the movement speed of the network node. Because the inputs and the outputs are characterized as discrete events, the traffic and the mobility models are designed with discrete-event system specification (DEVS) formalism [16].

For training of the communication effects, in the first two phases, we preferentially performed SoS-based simulation. The SoS simulation consists of two systems and an interoperable middleware between them: a combat system containing military operations of various combat entities, which is implemented through DEVSim++ [17], a network system that includes the depiction of a mobile ad hoc network (MANET) with network simulator 3 (NS3) [18, 19], and a runtime infrastructure (RTI) for interoperating them [20]. In this paper, we assume that the simulation results for SoS-based NCW are already validated.

As an experiment, we measured the accuracy and simulation speed of the procedure of transformation by comparing the previous traffic and mobility model in the existing study. The experimental result shows that the proposed work reduced the error by about 6.40% compared to the previous work and within an acceptable added time for training the proposed model. Finally, we expect that our study will provide an alternative way for the user, when operating an SoS requiring long execution times, to conduct a simulation analysis of various scenarios including network parameters such as a sensitivity analysis.

This study is organized as follows. Section 2 describes the background. Section 3 analyzes previous work and its limitations. Section 4 defines our problem and explains the proposed method and model. Section 5 discusses the

experimental results by comparing previous studies. Finally, Section 6 presents our conclusions.

2. Background

This section provides background knowledge regarding systematic views and the main factors affecting communication analysis in NCW.

2.1. Two Systematic Views for Communication Analysis in NCW. The power of NCW is derived from the effective linking or networking of knowledgeable entities that are geographically or hierarchically dispersed [21]. The combat entities in NCW can independently move in any direction; also, they are interconnected with wireless communication. Thus, we assume that the network for NCW is realized with MANET, which is effective in enabling highly mobile, highly responsive, and quickly deployable entities [22]. In this context, one of the purposes of NCW simulation is to analyze the performance of the network, for example, packet delivery ratio or end-to-end delay, in consideration of the network's dynamic configuration [23].

Figure 1 shows two systematic views on communication analysis in MANET. The first approach is based on the SoS-based simulation. In the SoS view, a combat system generates the tactical behaviors of all entities and a network system computes the configuration of nodes in MANET. Two main data sets, that is, traffic and mobility, are interacted with each other via a predefined interface. The next view is to develop an integrated network system comprising a network model and an abstracted combat model. The combat model abstracts mobility and traffic data and interacts with the network model at the model level not the system level.

The interoperation method assumes that each subsystem performs its tasks autonomously to take detailed actions for the SoS [11]. Thus, the biggest benefit of the SoS-based NCW is enhancing the accuracy of mobility and traffic data. Nevertheless, it inevitably remains a practical problem due to the prolonged simulation execution time. On the other hand, the integration method implies that each submodel interacts with others, sharing common information to form a unified system. Since they are operated within a standalone environment, the simulation is executed more quickly than in the interoperation method. However, the abstracted models have lower fidelity compared with the independent systems in the SoS.

Therefore, this paper proposes an advanced integration method for communication analysis in NCW, which reduces the simulation execution time and ensures the accuracy of the mobility and traffic models.

2.2. Two Factors for Communication Analysis in NCW: Mobility and Traffic. Figure 2 illustrates how two factors, that is, traffic and mobility, influence the linking of combat entities in NCW. We assume that there is a one-to-one correspondence between a combat entity and a network node. For a straightforward understanding, we explain this by separating

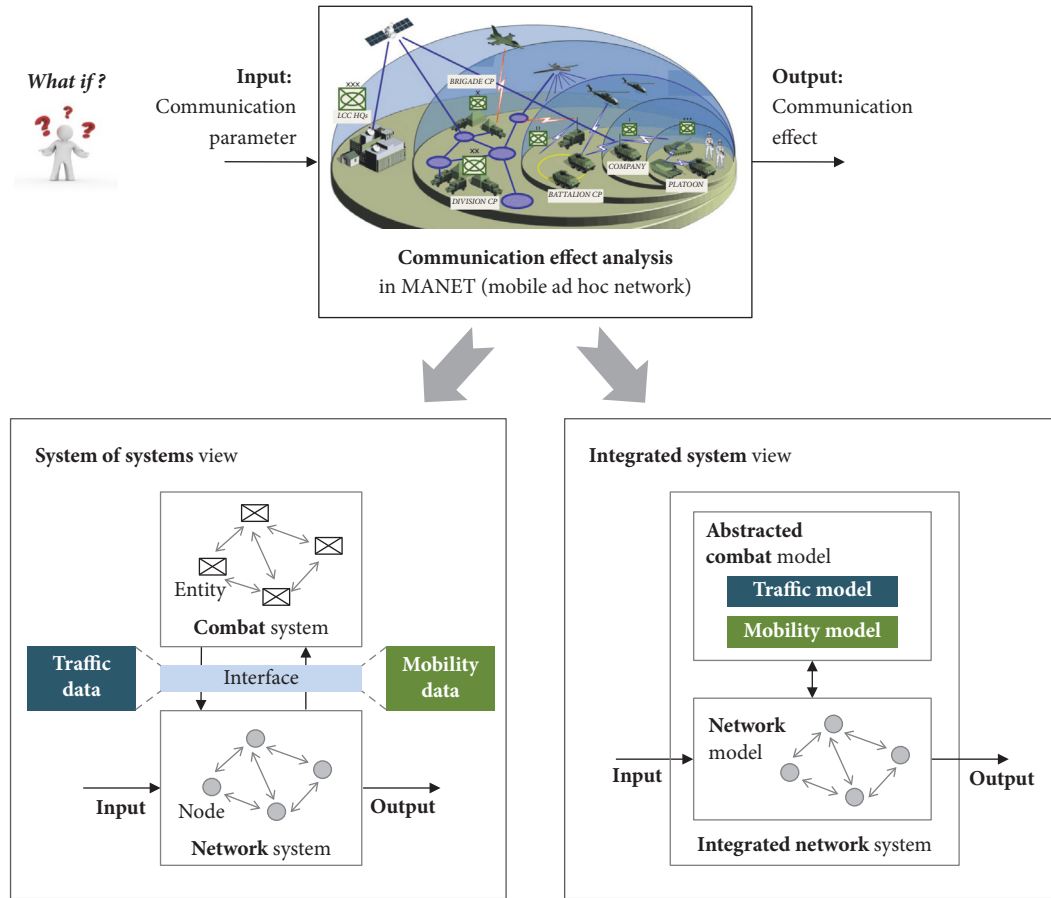


FIGURE 1: Communication analysis from two systematic perspectives: system of systems (SoS) and integrated system views.

the combat entities from their nodes, which is relevant to an SoS approach.

In Figure 2, a C2 center, for example, headquarters, tries to send an infantry troop a command message to move to a specific location. Under normal situations, Entity_{C2} in the combat system sends a request message, including the command to the network system. Then, the network system finds a proper routing path from Node_{C2} to Node_A and sends a response message to Entity_A in the combat system. After receiving the message, Entity_A moves to the desired location. On the contrary, when the network is in a bad condition, the transmission from Node_{C2} to Node_A fails, and the network system cannot send a response to the combat system. In this case, Entity_{C2} has difficulty controlling Entity_A, and Entity_A may generate different maneuvering behavior. If Entity_A moves, whether it behaves correctly or not, it sends a final position to the network system to update its node. In other words, this ad hoc configuration of the network may change with time as the nodes move or adjust their transmission and reception messages.

In this way, the combat system exchanges two types of data, that is, traffic and mobility, with the network via an interoperation infrastructure. The traffic data regarding the command message in Figure 2 is composed of the traffic flow in the connection of the pair of the source and the destination

entities and the requested time for the connection. The mobility data has the position of the entity at the specific time.

With these data, when we transform the SoS-based NCW simulation into the integrated simulation, including the combat and the network models, the following two requirements should be considered. First, traffic and mobility occur eventually in the combat model; thus, the two models are specified with discrete-event simulations. Next, because these data influence both models, it is necessary to construct them with inputs as well as outputs.

Although many research studies have been conducted on the construction of the traffic and mobility model in the network system [24, 25], these regard the models as a generator model with only output. In the following section, we will discuss some related work and its limitations in detail.

3. Related Works and Limitations

Over the last decade, several studies have been conducted to construct a traffic and mobility model for evaluation of the network system in the NCW. Each of these studies has described this model in the form of a generator that can be classified in terms of whether data is used in the process of the model's construction or not. The generator model not based

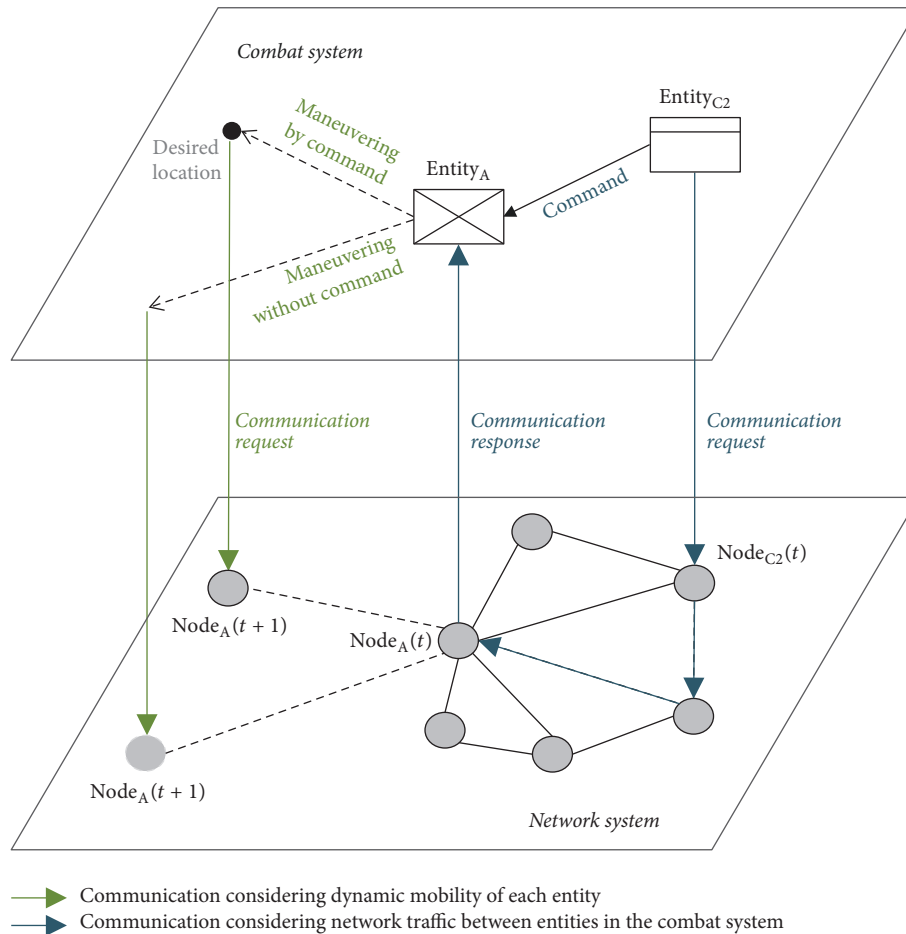


FIGURE 2: Traffic and mobility data exchanged between the combat system and the network system.

on the data has an advantage in that it can be used regardless of the presence or absence of data, whereas it is difficult to reflect the real world completely. On the other hand, the generator model based on data enhances the reality of the traffic and mobility, although it requires data from the real world. Table 1 shows the related traffic and mobility models according to the criteria for this classification.

To evaluate the performance of the network system, even though there is no available data, several studies have used the statistical traffic models based on probabilistic distribution, such as constant bit ratio and exponential. Some of them have represented the network with homogeneous nodes with the same traffic [26–28] or with heterogeneous nodes with different traffic according to the type of the combat entity [29]. On the other hand, other studies have tried to construct the traffic model with available data from the real world [30, 31]. They have extracted the data set of the interdeparture time from the real world and have made the empirical cumulative distribution function using the data set. They have then generated the traffic based on the distribution function. Although these models play a role in generating traffic, they cannot generate the different traffic against a change in the situation, because they are only in the form of a generator with the output.

Similar to the traffic model, some researchers have focused on developing a mobility model to evaluate the network system under a condition in which there is no available data. Bindra et al. and Kioumourtzis et al. used the reference point group mobility model that regards mobile nodes as a moving group based on a random waypoint model (RWP) [32, 33]. Also, Fongen et al. used the hierarchical group mobility model that divides the battlefield area according to the organization structure and moves within the area using the RWP model [34], while Reidt and Wolthuse used the ad hoc mobility model with the tactical maneuver trace predefined by the user [35]. Although their studies can be an alternative when there is no available data, the stereotyped model is somewhat different from the situation in the real world. On the other hand, other studies have tried to construct a similar mobility model that reflects the real world from the data [36, 37]. In these studies, they hypothesized a mobility model that includes the combat entity's property as the parameter, such as speed, the angle of movement, and the duration time of the movement, and then tunes the parameters based on the traced data acquired from the real world. In the end, two such models can only generate the predefined positions regardless of the change of the situation, because they are only generator models.

TABLE I: Related works for construction of the traffic and mobility model.

Model type	Pros and cons	Kinds of traffic and mobility model
Generator model not based on the data	Pros (i) Can be used as an alternative if there is no available traffic or mobility data Cons (i) Is limited in terms of expressiveness	Traffic model (i) Statistical traffic model [26–29] Mobility model (i) Reference point group mobility model [32, 33] (ii) Hierarchical group mobility model [34] (iii) Ad hoc user-defined model [35]
Generator model based on the data	Pros (i) Enhances the reality of the traffic and mobility Cons (i) Requires the traffic or mobility data for learning	Traffic model (i) Traffic model from data in the real world [30, 31] Mobility model (i) Mobility model from data in the real world [36, 37]

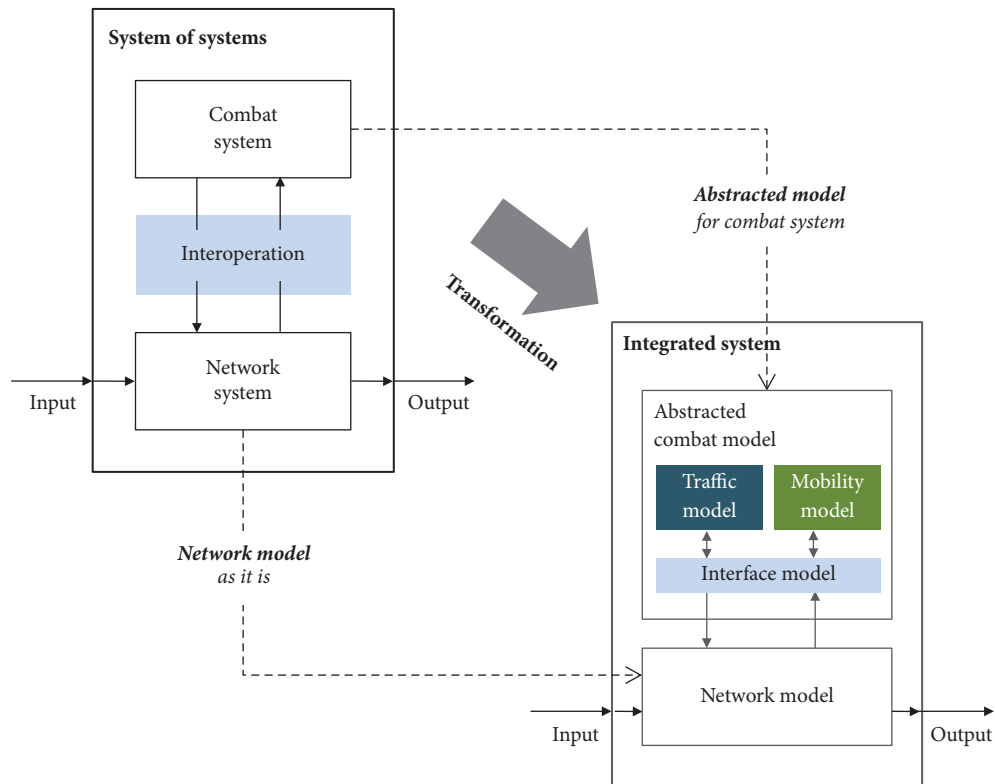


FIGURE 3: Transformation of the SoS into the integrated system for NCW simulation.

To summarize, even though some researchers have tried to construct the traffic and mobility models regardless of the presence or absence of data for the analysis of the NCW, they have represented the models in the form of a conventional generator, only having an output from the view of the standalone system and not the system of systems. Unfortunately, to apply these generator models to the analysis of the network system in the SoS-based NCW simulation causes the degradation of the accuracy, because the traffic and mobility data are changed according to the status of the communication, as we previously explained. For this reason, it is inevitable that we will need to construct a new type of traffic and mobility model that includes the input and output to generate the different traffic and mobility traces according to the status of the network system. Therefore, we suggest a

new type of traffic and mobility model that satisfies the above constraints of the previous studies. The following section focuses on this proposed model.

4. Proposed Work

In this section, we clarify the problem definition and propose a transformation method for NCW simulation. The proposed model in the method is an NCW model including an abstracted combat model and network model.

4.1. Problem Definition. Figure 3 shows how this study transforms the SoS based on an interoperation environment (upper left part) to the integrated system (lower right part). In this process, the network system remains as it is, whereas

the combat system is abstracted into an abstracted combat model, which includes the traffic and mobility generating the traffic and position data, and an interface model connecting the models with the network model. Then, the model is integrated with the unchanged network model.

In the process of the abstraction of the combat system, to deal with the aforementioned limitations, the proposed abstracted combat model should satisfy two requirements: (1) a discrete-event model for discrete-event simulation and (2) a model generating the different output according to the states of the adjacent system. In this respect, we define such a situation as a problem to (1) hypothesize a DEVS coupled model including atomic models and (2) estimate the related variables.

4.2. Proposed Transformation Method. In general, the output of the system ($Y = f(X; V)$) can be expressed as a function (f) of input, that is, parameters (X) and variables (V). From this perspective, Figure 4 shows how formalism expresses the transformation, where

f is the combat system to be abstracted

g is the network system to be analyzed

X_g is the input parameter set of g ;

Y_g is the output set of g ;

V_g is the variable set of g ;

Y_{fg} is the output set from f to g ;

Y_{gf} is the output set from g to f ;

\widehat{f}_g is the abstracted model of f from the perspective of g ;

V_{fg} is the variable set of \widehat{f}_g ;

\widehat{h} is the estimation function for V_{fg} ;

$\widehat{Y}_{fg} = \widehat{f}_g(\widehat{Y}_{gf}; V_{fg})$ is the estimated output from \widehat{f}_g ;

$V_{fg} = \widehat{h}(\widehat{Y}_{gf})$ is the estimated variable of the model (\widehat{f}_g);

$\widehat{Y}_g = g(X_g, \widehat{f}_g(\widehat{Y}_{gf}; \widehat{h}(\widehat{Y}_{gf})); V_g)$ is the integrated system.

In the SoS, the output of the network system is $Y_g = g(X_g, Y_{fg}; V_g)$, and Y_{fg} is acquired from the combat system ($Y_{fg} = f(Y_{gf}; V_f)$). Unfortunately, because the combat system does not exist in the integrated system, Y_{fg} cannot be acquired directly.

To solve this problem, we hypothesize an abstracted model ($\widehat{Y}_{fg} = \widehat{f}_g(\widehat{Y}_{gf}; V_{fg})$). \widehat{f}_g and V_{fg} refer to an abstracted combat model and a variable of the combat model from the perspective of the analysis of the network system. For example, among the various combat system logics, factors only related to the traffic and position are necessary for

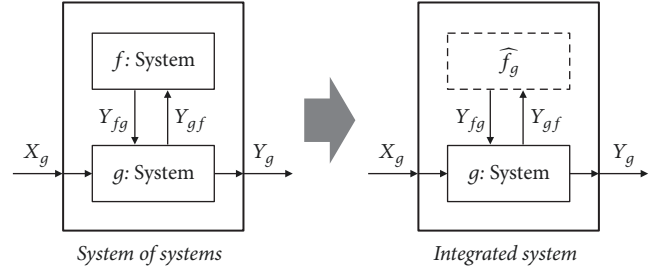


FIGURE 4: Systematic representation of the proposed transformation.

the network system's analysis. In this context, V_{fg} means the variable for expressing the factors and \widehat{f}_g means the abstracted combat model, including the variable. In addition, to represent the fact that V_{fg} is influenced by the response of the network system (i.e., \widehat{Y}_{gf}), we assume an estimation function (\widehat{h}) against the relation between V_{fg} and \widehat{Y}_{gf} ; that is, $V_{fg} = \widehat{h}(\widehat{Y}_{gf})$.

With \widehat{f}_g and \widehat{h} , this paper can acquire the output, $\widehat{Y}_{fg} = \widehat{f}_g(Y_{gf}; \widehat{h}(\widehat{Y}_{gf}))$, from the abstracted combat model, and furthermore, by using this model, we can finally find the output of the integrated network system $\widehat{Y}_g = g(X_g, \widehat{f}_g(\widehat{Y}_{gf}; \widehat{h}(\widehat{Y}_{gf})); V_g)$ by integrating the network model. The next subsection shows its detailed process.

4.3. Overall Process of System Transformation. Figure 5 depicts the process of the transformation consisting of the following three major phases: (1) data acquisition from the simulation of the SoS-based NCW, (2) data preprocessing, and (3) traffic and mobility model hypothesis and variable estimation of the models using the neural network. In the model hypothesis step, Figure 6 and the following specification indicate an abstracted combat model (ACM) constructed as a DEVS coupled model, including three kinds of DEVS atomic models: traffic (TM), mobility (MM), and interface model (IM). The role of the traffic model is to generate data from the source node to the destination node; each traffic model is mapped onto the node's pair of the source node and the destination node. The mobility model generates the position of the corresponding node; each model is mapped to each node.

In the last phase, we hypothesized the traffic and mobility model and the function for estimating the variables of the models using the neural network; we trained the neural network using the preprocessed data set. Then we constructed the integrated system for the NCW by embedding the neural network in the traffic and mobility model. The next subsection gives a description of the last phase.

4.4. Proposed NCW Model. In the first phase, we extracted some experimental points against the input of the network system (i.e., communication parameters) from the entire design space using the design of the experiment (DOE). We

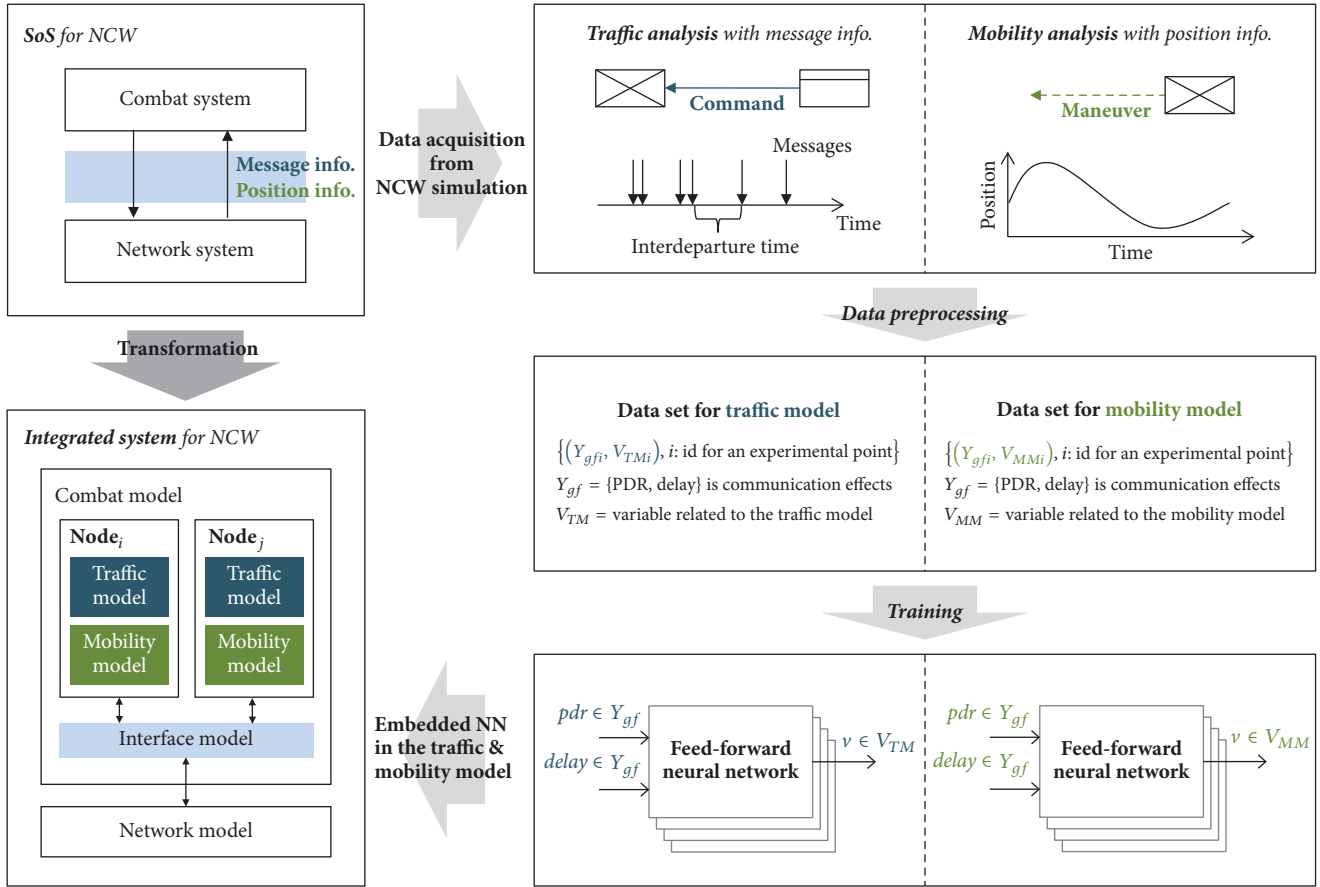


FIGURE 5: Proposed transformation procedure of the SoS into the integrated system.

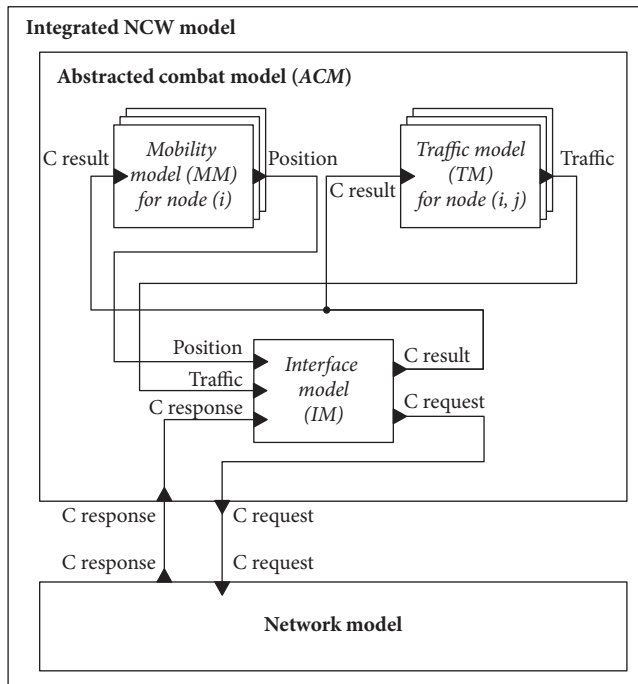
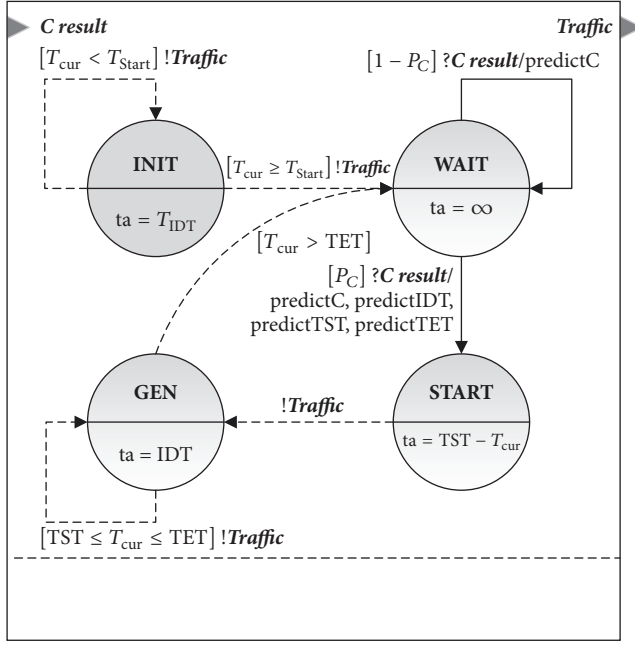


FIGURE 6: Overall NCW model structure including the abstracted combat model and the network model.

then executed the NCW simulation against the extracted experimental points by acquiring the message and position information for the traffic and mobility analysis. After that, for the preprocessing of the data for training, we constructed the data set for traffic and mobility from the acquired information against the experimental points. The former is $\{(Y_{gfi}, V_{TMi}), i: \text{id for an experimental point}\}$ and the latter is $\{(Y_{gfi}, V_{MMi}), i: \text{id for an experimental point}\}$, where Y_{gfi} is communication response, that is, end-to-end delay and PDR, and V_{TMi} and V_{MMi} are the variables related to the traffic and mobility model, respectively.

In addition, to represent the property of the traffic and mobility affected by the network system, the interface model performs a role to connect the communication effects from the network model to the input of the traffic and mobility model according to the following procedure. The interface model transmits the traffic generated from the traffic model to the network model through the C request during the initial period of the simulation. The model then receives the communication effect of each event from the network model through the C response port, calculates the average communication effect, and transmits it to the traffic and mobility model using the C result port. Once it has received the effects, the traffic and mobility model completes the simulation model by calculating the value of the variables of the model based on the effects



→ External transition, Transition Syntax: [Condition] ?input/
Object model
--> Internal transition, Transition Syntax: [Condition] !output/
Object model

FIGURE 7: Traffic model design using the DEVS formalism.

and the neural network and proceeds with the simulation.

$$ACM = \langle X, Y, \{M_i\}, EIC, EOC, IC, sel \rangle, \quad (1)$$

where

$$\begin{aligned} X &= \{C \text{ response}\}; \\ Y &= \{C \text{ request}\} \\ \{M_i\} &= \{\bigcup_{i=1}^n TM_i, \bigcup_{i=1}^n MM_i, IM\}; \\ EIC &= \{(ACM.C \text{ response}, IM.C \text{ response})\}; \\ EOC &= \{(IM.C \text{ result}, ACM.C \text{ request})\}; \\ IC &= \{(IM.C \text{ result}, \bigcup_{i=1}^n TM_i.C \text{ result}) \\ &\quad (IM.C \text{ result}, \bigcup_{i=1}^n MM_i.C \text{ result}), \\ &\quad (\bigcup_{i=1}^n TM_i.Traffic, IM.Traffic) \\ &\quad (\bigcup_{i=1}^n MM_i.Position, IM.Position)\}; \\ sel &= IM. \end{aligned}$$

Figure 7 and the following specification show the structure of the traffic DEVS atomic model (TM) consisting of four variables: $V_{TM} = \{IDT, P_C, TST, TET\}$. IDT and P_C refer to the time interval between the generation of the traffic and whether or not the source and destination node are connected. TST and TET are the times when the packet generation is started and ended.

$$TM = \langle X, Y, S, \delta_{ext}^p, \delta_{int}, \lambda, ta \rangle, \quad (2)$$

where

$$X = \{C \text{ result}\};$$

$$Y = \{Traffic\}$$

$$S = \{INIT, WAIT, START, GEN\}$$

$$\delta_{ext}^p : (WAIT, C \text{ result}, P_C) \rightarrow START$$

$$\begin{aligned} &\text{execute } \forall om \in OM_{TM}; \\ &(WAIT, C \text{ result}, 1 - P_C) \rightarrow WAIT \\ &\text{execute } predictP_C; \end{aligned}$$

$$\delta_{int} : (START, TST - T_{cur}) \rightarrow GEN$$

$$(GEN, IDT) \rightarrow GEN \text{ where } TST \leq T_{cur} \leq TET$$

$$(GEN, IDT) \rightarrow WAIT; \text{ where } TET < T_{cur}$$

$$(INIT, T_{IDT}) \rightarrow INIT; \text{ where } T_{cur} < T_{START}$$

$$(INIT, T_{IDT}) \rightarrow WAIT; \text{ where } T_{START} \leq T_{cur}$$

$$\lambda : \{START, GEN, INIT\} \rightarrow Traffic$$

$$ta : INIT \rightarrow T_{IDT};$$

$$WAIT \rightarrow \infty;$$

$$START \rightarrow TST - T_{cur}$$

$$GEN \rightarrow IDT.$$

$$V_{TM} = \{IDT, P_C, TST, TET\} \quad (3)$$

is TM 's variable:

IDT is the interdeparture time of traffic generation;

P_C is the probability of existence of the traffic model;

TST is the start time of the traffic generation;

TET is the end time of the traffic generation.

$$OM_{TM} = \{predictIDT, predictP_C, predictTST, predictTET\} \quad (4)$$

is V_{TM} 's object model:

$predictIDT$ is the prediction function of IDT ;

$predictP_C$ is the prediction function of P_C ;

$predictTST$ is the prediction function of TST ;

$predictTET$ is the prediction function of TET ;

T_{START} is the time for the initial traffic generation;

T_{IDT} is the interdeparture time of the traffic in the $INIT$ state.

The traffic model is a probabilistic discrete-event model in which packets are generated at intervals of IDT from the TST to the TET time according to the probability of the existence of the connection (P_C) [38]. The model has four states: $INIT$, $WAIT$, $START$, and GEN . In the $INIT$ state, traffic occurs in the T_{IDT} interval time during

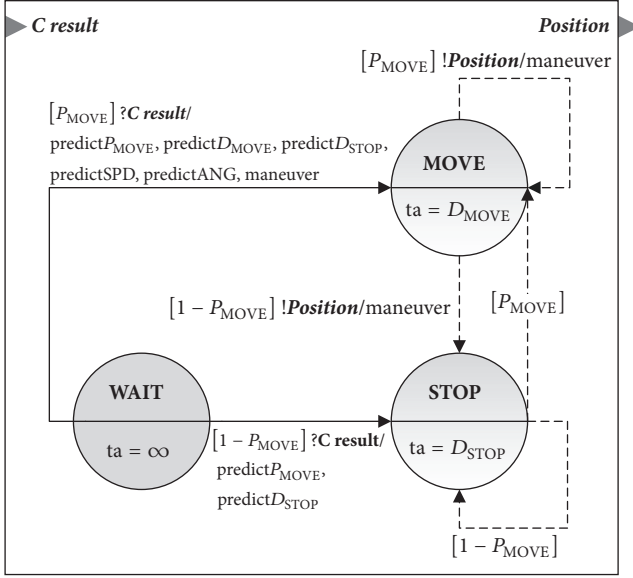


FIGURE 8: Mobility model design using the DEVS formalism.

the T_{START} time and then waits in the *WAIT* state. After receiving the communication effect from the interface model through the *C result*, the IDT , P_C , TST , and TET values are calculated by calling the object model: $OM_{TM} = \{predictIDT, predictP_C, predictTST, predictTET\}$. If the connection exists, it transits to the *START* state and to the *GEN* state after the $TST - T_{cur}$ time for traffic generation and generates the traffic at the IDT interval. Then, when the TET time is reached, the traffic generation is stopped and the state transits to the *WAIT* state. On the other hand, if *C* does not exist, it does not generate traffic. In such a form, this traffic model plays the role of generating different types of traffic according to the communication effect calculated in the network model.

Figure 8 and the following specification show the structure of the mobility DEVS atomic model (*MM*) consisting of five variables: $V_{MM} = \{P_{MOVE}, D_{MOVE}, D_{STOP}, SPD, ANG\}$. P_{MOVE} refers to the transition probability to the *MOVE* state, and D_{MOVE} and D_{STOP} refer to the time for remaining in the *MOVE* and *STOP* state, respectively. In this model, the probability of staying in the *MOVE* state is determined by P_{MOVE} , and the position information is generated based on the SPD and ANG in that state.

$$MM = \langle X, Y, S, \delta_{ext}^p, \delta_{int}^p, \lambda, ta \rangle, \quad (5)$$

where

$$X = \{C \text{ result}\}$$

$$Y = \{Position\};$$

$$S = \{WAIT, MOVE, STOP\}$$

$$\delta_{ext}^p : (WAIT, C \text{ result}, P_{MOVE}) \rightarrow MOVE$$

$$\begin{aligned} & \text{execute } \forall om \in OM_{MM} \\ & (WAIT, C \text{ result}, 1 - P_{MOVE}) \rightarrow STOP \end{aligned}$$

execute $predictP_{MOVE}, predictD_{STOP}$

$$\delta_{int}^p : (MOVE, D_{MOVE}, P_{MOVE}) \rightarrow MOVE$$

execute *maneuver*

$$(MOVE, D_{MOVE}, 1 - P_{MOVE}) \rightarrow STOP$$

execute *maneuver*

$$(STOP, D_{STOP}, P_{MOVE}) \rightarrow MOVE$$

$$(STOP, D_{STOP}, 1 - P_{MOVE}) \rightarrow STOP$$

$$\lambda : MOVE \rightarrow Position;$$

$$ta : WAIT \rightarrow \infty;$$

$$MOVE \rightarrow D_{MOVE};$$

$$STOP \rightarrow D_{STOP}.$$

$$V_{MM} = \{P_{MOVE}, D_{MOVE}, D_{STOP}, SPD, ANG\} \quad (6)$$

is *MM*'s variable:

P_{MOVE} is the transition probability to the *MOVE* state;

D_{MOVE} is the duration time in the *MOVE* state;

D_{STOP} is the duration time in the *STOP* state;

SPD is the speed of the movement;

ANG is the angle of direction change.

$$\begin{aligned} OM_{MM} = \{ & predictP_{MOVE}, predictD_{MOVE}, \\ & predictD_{STOP}, predictSPD, predictANG, \\ & maneuver \} \end{aligned} \quad (7)$$

is V_{MM} 's object model:

$predictP_{MOVE}$ is the prediction function of P_{MOVE} ;

$predictD_{MOVE}$ is the prediction function of D_{MOVE} ;

$predictD_{STOP}$ is the prediction function of D_{STOP} ;

$predictSPD$ is the prediction function of SPD ;

$predictANG$ is the prediction function of ANG ;

maneuver is the maneuver function using SPD, ANG .

This model is a probabilistic discrete-event model in which the model varies stochastically according to the P_{MOVE} . The model has three states, *WAIT*, *MOVE*, and *STOP*, and waits initially in the *WAIT* state. Similar to the traffic model, after receiving input through the *C result*, five parameter values are calculated by the object model: $OM_{MM} = \{predictP_{MOVE}, predictD_{MOVE}, predictD_{STOP}, predictSPD, predictANG, maneuver\}$. In the *MOVE* state, the position is updated through the SPD and ANG during the D_{MOVE} time, but the position is not updated during the D_{STOP} time in the *STOP* state. At the end of each state, it decides the probability of staying in the *MOVE* state according to P_{MOVE} .

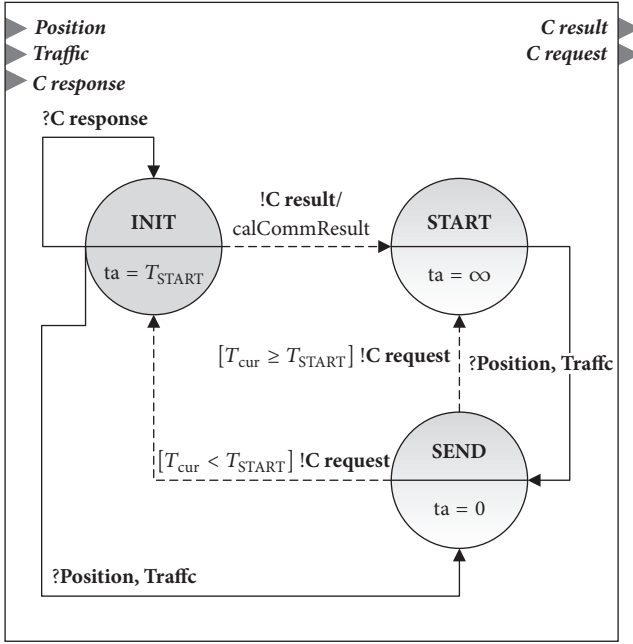


FIGURE 9: Interface model design using the DEVS formalism.

In this way, the influence of the network model affects the values of the five variables through the object model, thereby generating different types of mobility.

Finally, the specification for the interface DEVS atomic model (*IM*) is described in Figure 9. This model calculates the cumulative communication effect through the *C response* from the network model during the initial T_{START} time and passes the result to the traffic and mobility model through the *C result* port. Then, the traffic and position information generated from each model is transmitted to the network model through the *C request* port. The interface model consists of three states: *INIT*, *START*, and *SEND*. After the T_{START} time in the *INIT* state, the model calculates the average communication effects using the *calCommResult* and transmits them to the traffic and mobility model. Through this process, the model enables the generation of the different types of traffic and mobility data according to the state of the network model.

$$IM = \langle X, Y, S, \delta_{ext}, \delta_{int}, \lambda, ta \rangle, \quad (8)$$

where

$$X = \{Position, Traffic, C\ response\}$$

$$Y = \{C\ result, C\ request\};$$

$$S = \{INIT, START, SEND\}$$

$$\delta_{ext} : (INIT, C\ response) \rightarrow INIT$$

$$(INIT, Position\ or\ Traffic) \rightarrow SEND$$

$$(START, Position\ or\ Traffic) \rightarrow SEND$$

$$\delta_{int} : INIT \rightarrow START;$$

$$SEND \rightarrow INIT\ where\ T_{cur} < T_{START}$$

$$SEND \rightarrow START\ where\ T_{START} \leq T_{cur}$$

$$\lambda : INIT \rightarrow C\ result;$$

execute *calCommResult*

$$SEND \rightarrow C\ request;$$

$$ta : INIT \rightarrow T_{START};$$

$$START \rightarrow \infty;$$

$$SEND \rightarrow 0.$$

calCommResult is the function for calculation on the effects of communication.

After the model hypothesis step, we are going to focus on the variable (V_{TM}, V_{MM}) estimation step. As mentioned in the previous subsection, it is necessary to transform raw data $\{Y_{gf}, Y_{fg}\}$ from the SoS-based NCW simulation to the data set $\{Y_{gf}, V_{fg}\}$ for learning (referring to Figure 5). We then constructed a regression model with the form of feed-forward neural network using the data set [39–41].

The upper part of Figure 10 shows the detailed structure including Y_{gf} as an input and V_{fg} as an output. It consists of three kinds of layers: input, hidden, and output; each layer has neurons, and they have a weighted connectivity with neurons in the other layers. The value of neurons in the hidden and output layer is acquired from the weighted sum of neurons in the previous layer and the activation function. To make this identified regression model in an executable form, as shown in the lower part of Figure 10, this paper regards the regression model as a function and generates the model to the source codes, which can be implemented to the prediction functions of the object model (OM_{TM}, OM_{MM}) in the above traffic and mobility model.

5. Case Study: Simulation-Based Analysis of the Network System

The objective of this case study is to demonstrate how much the proposed method improves the accuracy compared to the existing method while conducting the transformation of the SoS-based NCW [42, 43]. Also, this paper compares the simulation speeds before and after the transformation.

5.1. Experimental Design. The objective of the SoS-based NCW simulation is to analyze the communication effects (i.e., end-to-end delay and PDR) against the communication-related parameters under the battlefield environment in which the exchange of information between combat entities occurs through the communication. The SoS-based NCW consists of two systems (combat and network system), which have already been validated.

The combat system represents the army's military logic at an infantry company level and consists of 131 combat entities. These entities exchange information between entities through communication (traffic) and conduct the maneuver (mobility). This system is implemented in the DEVSIMHLA using DEVS formalism. On the other hand, the network

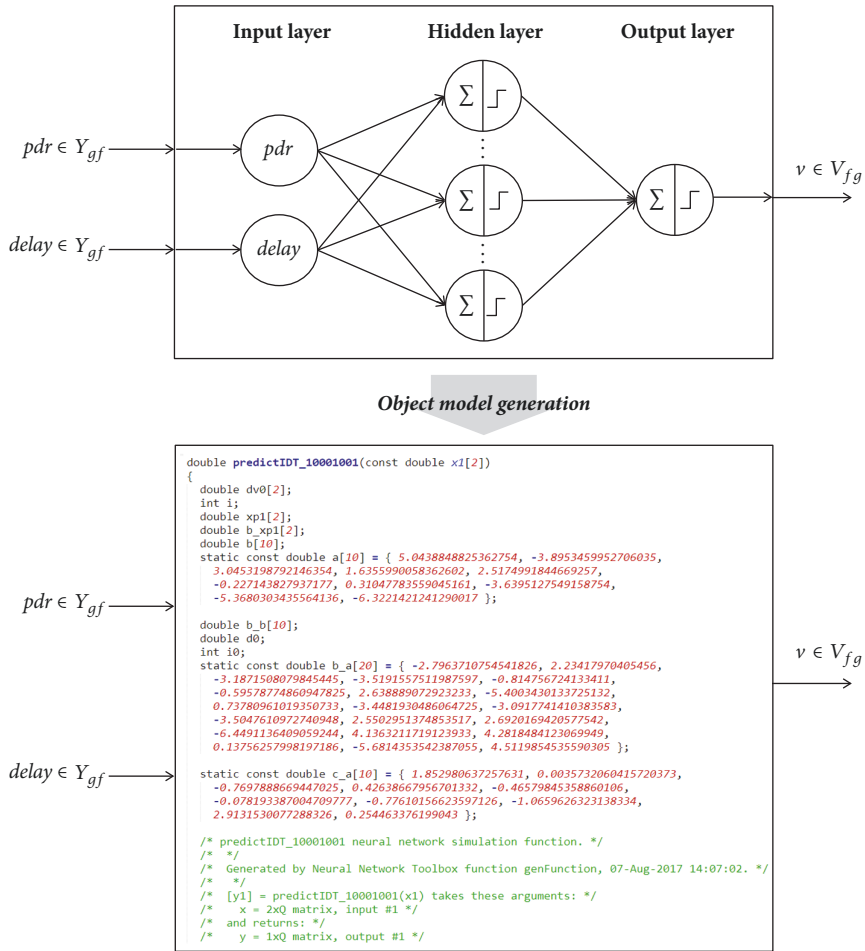


FIGURE 10: Realization of the object model from the trained feed-forward neural network.

TABLE 2: Network system parameters and their description.

Parameter name	Parameter level	Description
Packet size (P_{PS})	100, 200, 400, ..., 6400 (byte)	The size of packet
Transmission power (P_{TP})	-10, 5, 0, ..., 40 (dbm)	The transmission power of node
Transmission gain (P_{TG})	0, 2, 4, ..., 20 (dB)	The transmission gain of node
Reception gain (P_{RG})	0, 2, 4, ..., 20 (dB)	The reception gain of node
PhyMode (P_{PM})	1, 2, 5.5, 11 (Mbps)	The 802.11 phy layer mode of DsssRate

system describes the MANET using destination sequenced distance vector (DSDV) routing protocol and consists of 131 network nodes, which correspond to the entities of the combat system [44, 45]. This system calculates the effects of communication (traffic) by updating the position of nodes (mobility). This system is implemented in the ns-3 discrete-event network simulator.

The two systems participate in the HLA-based interoperable simulation through an RTI and use a federation object model that includes the traffic and mobility data. In addition to the data management, the systems advance the simulation time using the application program interfaces related to time management. The network system calculates the effects of communication against the traffic data from the

combat system and transmits them to the destination node by updating information on the position from the mobility data.

This combat scenario depicts the complex and hierarchical information exchange among combat entities [46, 47]. The blue force conducts a defense operation against the red force with three times the military strength in a 2 km \times 2 km operation area. The combat entities perform threat evaluation and weapons assignment and transmit the results to another entity through communication. According to the communication performance, the entity makes a different decision and generates a different type of traffic and mobility data. Table 2 shows the parameters' names and their descriptions.

TABLE 3: Linear regression model of packet delivery ratio.

	Coefficient	SE	t Stat	p value
C	0.3844	0.0498	7.7202	$1.63e - 10$
P_{PS}	$-1.59e - 05$	$7.00e - 06$	-2.2756	0.0265
P_{TP}	0.0102	$9.4723e - 04$	10.7808	$1.43e - 15$
P_{TG}	0.0088	0.0024	3.7153	$4.5282e - 04$
P_{RG}	0.0084	0.0024	3.5254	$8.2480e - 04$
P_{PM}	-0.0160	0.0043	$-3.6865e - 04$	$4.9648e - 04$

Adj. $R^2 = 0.712$; p value = $8.14e - 16$.

TABLE 4: Linear regression model of end-to-end delay.

	Coefficient	SE	t Stat	p value
C	0.0775	0.0660	1.1737	0.2452
P_{PS}	$4.68e - 05$	$9.28e - 06$	5.0439	$4.64e - 06$
P_{TP}	0.0033	0.0013	2.6683	0.0098
P_{TG}	$9.3956e - 04$	0.0031	0.2987	0.7662
P_{RG}	$6.5681e - 04$	0.0031	0.2088	0.8354
P_{PM}	-0.0234	0.0057	-4.0851	$1.34749e - 04$

Adj. $R^2 = 0.39$; p value = $1.66e - 06$.

5.2. Experimental Procedure. Before conducting the transformation, we executed the SoS-based NCW simulation. In our five-dimensional network parameters in Table 2, the full factorial design size is 7 (for P_{PS}) \times 11 (for P_{TP}) \times 11 (for P_{TG}) \times 11 (for P_{RG}) \times 4 (for P_{PM}) = 37,268, which requires a long execution time, approximately 1,128,084 hours if we conduct 30 trials against each experimental point (1.01 hours per one execution) [48–50]. For this reason, we selected the first 65 experimental points for training from the entire design space: 43 points using face-centered central composite and 22 points using a Latin hypercube design [51]. We then chose an extra 22 points at random from among the full design to evaluate the transformation. Using the acquired data from the SoS-based NCW simulation, we constructed an abstracted combat model consisting of the 908 traffic models, 131 mobility models, and an interface model. In the process, we trained the feed-forward neural network with a 2-5-1 structure using Levenberg-Marquardt algorithm.

From the perspective of accuracy and simulation execution performance, we regarded the output of the network system and the execution time as the effectiveness index. The environment for this case study is as follows. For the combat system, CPU: I5-3550 3.3 GHz, RAM: 4 GB, DEVSim++ v.3.1 are used. For the network system, we used NS3 v.3.18. These two systems used RTI 1.3-NG and progressed simulation time over 5,000 sec including T_{START} 300 sec. For the training, we used MATLAB neural network toolbox v.8.2.1 and MATLAB Coder v. 2.7.

5.3. Experimental Results. Before the analysis of the accuracy and the speed, we constructed the first-order linear regression model ($y \sim 1 + P_{PS} + P_{TP} + P_{TG} + P_{RG} + P_{PM}$) to identify whether the selected network parameters influence the communication effects [52, 53]. In Tables 3 and 4, the column refers to the parameters, while the row refers to

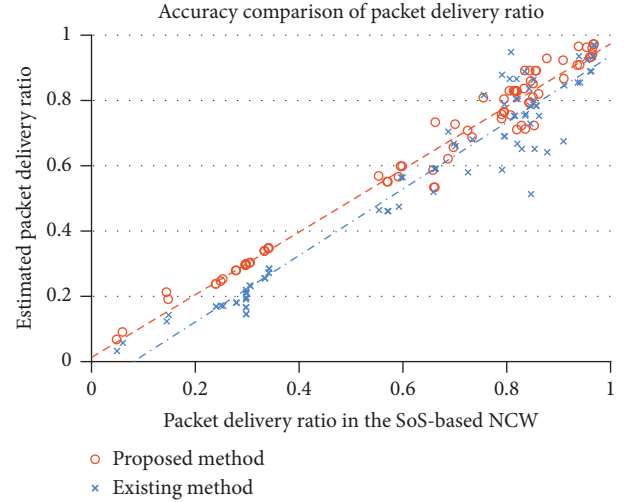


FIGURE 11: Simulation results for packet delivery ratio comparing between the existing and the proposed methods.

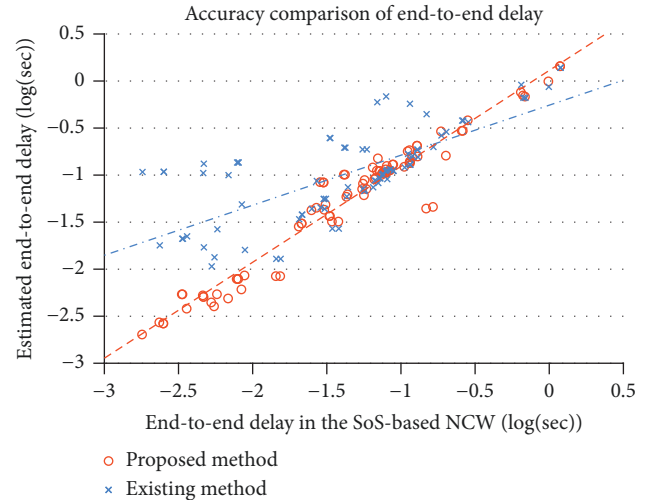


FIGURE 12: Simulation results for end-to-end delay comparing between the existing and the proposed methods.

the coefficient estimates, the standard errors (SEs) of the estimates, the t -statistic values of the hypothesis tests for the corresponding coefficients (t Stat), and the significant probability (p value). Tables 3 and 4 show that the five parameters influence at least one of the two communication effects based on the fact that the p value is smaller than the significance level (0.05); therefore, we used these parameters in the case study.

To conduct an analysis from the perspective of accuracy, we compared the results of the transformation through the proposed method and the conventional method, which only generates the output regardless of the input. Figures 11 and 12 show the graphs for the accuracy comparison of the communication effects (i.e., PDR and end-to-end delay). The x -axis and y -axis refer to the communication effects from the SoS-based NCW execution and integrated system, respectively. The more symmetry of the x - and y -axes, the

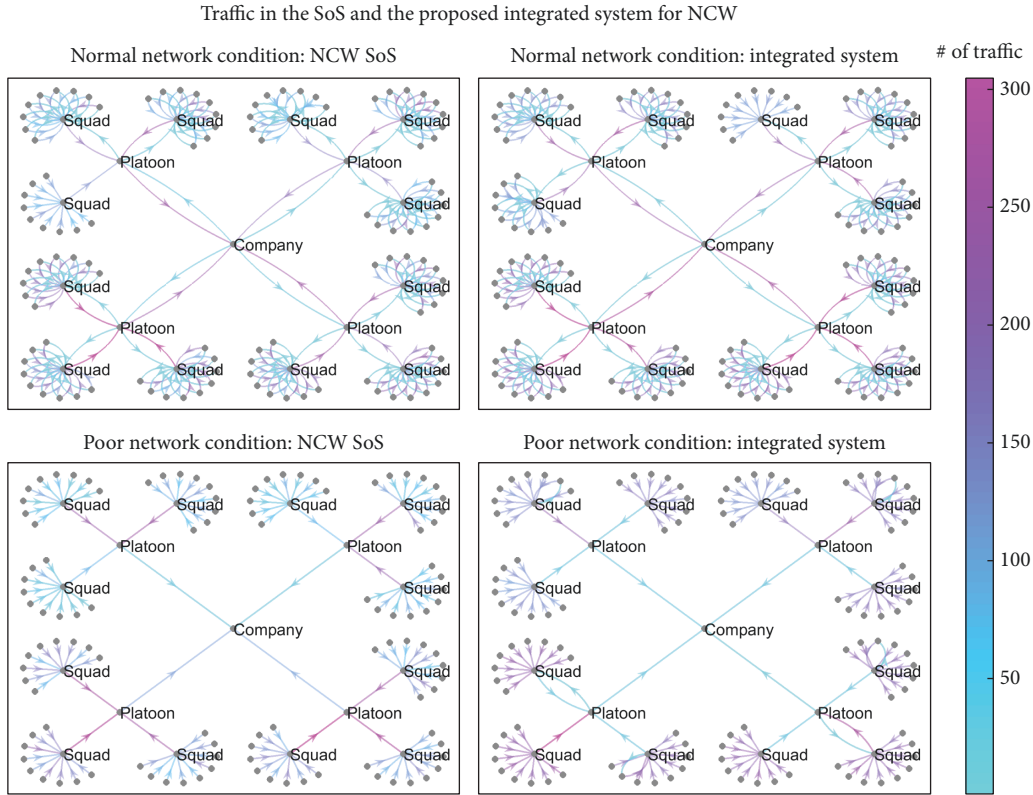


FIGURE 13: Traffic model results of the SoS and the proposed integrated system for NCW.

higher the accuracy. In Figures 11 and 12, the simulation results of the proposed method have higher accuracy than those of the conventional method. For the quantitate analysis of this difference, we measured the root mean square error (RMSE). In the case of Figure 11, each case indicates 0.0425 and 0.0997 RMSE, which means 4.6281% and 10.8407% when considering the minimum and maximum value (0.0491, 0.9691). Also, Figure 12 indicates 0.0520- and 0.1304-second RMSE, which means 4.3721 and 10.9647% when considering the minimum and maximum value (0.0018, 1.1915). This improved accuracy stems from the proposed traffic and mobility model, as shown in the following figures.

Figure 13 shows the flow of the traffic, which is one of the causes of enhanced accuracy. The left and right part show the traffic between entities in the SoS-based NCW and the proposed integrated system. The upper and lower part show the traffic in the normal and poor network condition; the former implies a case of having high PDR and small end-to-end delay, and the latter implies a case of having small PDR and high end-to-end delay. In the left part, the two figures show that more traffic occurs in the normal communication condition than in the poor condition, because the normal network condition allows for more exchanged information and more connection of information owing to the delivery of the hierarchical command. From this perspective, the right part describes a similar trend to the left part in that traffic changes according to the communication condition, which helps to reduce the error from the transformation.

TABLE 5: Comparison of the simulation speeds of the SoS and the integrated system for NCW.

	Execution time (min.)	# of processed events in the network system
SoS	60.539	2.539e + 08
Integrated system	16.025	2.569e + 08

Figure 14 shows another cause of enhanced accuracy, mobility data; it shows the average position change of the nodes against the experimental points at the end of the simulation time. In the SoS-based NCW simulation, the position change in the experimental points with a normal communication condition is larger than the experimental points with a poor condition, as the orders between the entities related to maneuver are normally transmitted through the communication. In the proposed integration system, the mobility also indicates a similar trend, although it does not have the same results; it also plays a role in enhancing the accuracy.

From the perspective of the simulation speed, we compared the average execution times and the average numbers of executed events per one trial in the SoS-based NCW and integrated system. Table 5 shows that the integrated system reduced the execution time 3.78-fold compared to the SoS-based NCW. Also, judging by the fact that the

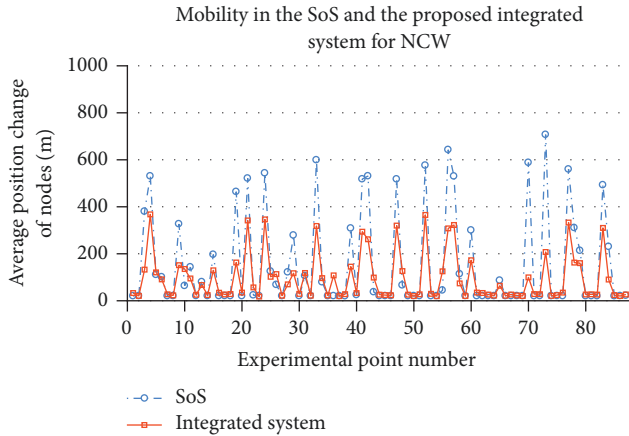


FIGURE 14: Mobility model results of the SoS and the proposed integrated system for NCW.

number of executed events in the network system is similar between the SoS-based NCW and the integrated system and is much higher than the number of the executed events in the combat system of the SoS-based NCW, we can infer that the elimination of the interoperation architecture of the SoS-based NCW plays a prominent role in reducing the execution time.

Furthermore, they recorded 2633.45, 697.07 hours with the 87 experiment points, that is, 2610 trials. In addition to the execution time for the simulation, the proposed integrated system requires a time for training in machine learning of about 8.55 hours, including 7.27 hours for the traffic models and 1.28 hours for the mobility models. Fortunately, however, the time is quite small compared to the execution time of the SoS-based NCW simulation.

6. Conclusion

In network-centric warfare (NCW), due to the importance of communication, which is responsible for the flow of information, it is necessary to analyze the performance of this communication against the communication parameters in an environment with high complexity, such as a battlefield. For this reason, many studies have conducted a simulation-based analysis of the NCW from the perspective of the SoS, which consists of the combat system and the network system so that the two reflect each other's effects. However, this paradoxically causes a prolonged execution time and difficulty in conducting the analysis of the various parameters due to the problem of time.

To overcome this weakness, we need to abstract the combat system to an abstracted combat model that includes the traffic and mobility models required for the network system's analysis. We also need to integrate the abstracted combat model with the network system. Some studies have been conducted on the construction of the traffic and mobility model for the analysis of communication. However, as the analysis has been performed in a standalone system, not an SoS, the resulting model has a form with only an output,

and therefore the model cannot generate the different outputs according to the state of the adjacent system, although it is an important characteristic of the SoS with high complexity.

This paper proposed the transformation of the SoS-based NCW into an integrated system. For this, we first hypothesized an abstracted combat coupled model that includes the traffic, mobility, and interface atomic models using discrete-event systems specification (DEVS) formalism for the discrete-event simulation. We then estimated the variables of the models in the form of the neural network, which can be updated from the state of the adjacent system, and identified the variable using the machine learning and the data acquired from the SoS-based NCW simulation execution.

The case study shows that the integrated system, as a result of the proposed method, significantly improves the accuracy compared to the existing method and reduces the execution time compared to the SoS-based NCW simulation. We expect that this paper will help in the analysis of various parameters in various domains based on the SoS as well as the military domain.

Glossary

C2:	Command and control
CPU:	Central processing unit
DEVS:	Discrete-event system specification
HLA:	High-level architecture
MANET:	Mobile ad hoc network
M&S:	Modeling and simulation
NCW:	Network-centric warfare
NS3:	Network simulator 3
PDR:	Packet delivery ratio
RAM:	Random-access memory
RMSE:	Root mean square error
RTI:	Run-time infrastructure
RWP:	Random waypoint model
SE:	Standard error
SoS:	System of systems.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korean Government (MSIP) (no. 2017-0-00461, Development Platform for User-Level Customizable, General Purpose Discrete-Event Simulation Software).

References

- [1] J. Moffat, "Complexity Theory and Network Centric Warfare," Defense Technical Information Center, 2003.
- [2] B. Xiong, B. Li, R. Fan, Q. Zhou, and W. Li, "Modeling and simulation for effectiveness evaluation of dynamic discrete

- military supply chain networks,” *Complexity*, Art. ID 6052037, 9 pages, 2017.
- [3] J. Cares, *Distributed networked operations: The foundations of network centric warfare*, iUniverse, 2005.
 - [4] C. K. Pang and J. Mathew, “Dynamically reconfigurable command and control structure for network-centric warfare,” *Simulation*, vol. 91, no. 5, pp. 417–431, 2015.
 - [5] D. R. Choffnes and F. E. Bustamante, “An integrated mobility and traffic model for vehicular wireless networks,” in *Proceedings of the VANET - 2nd ACM International Workshop on Vehicular Ad Hoc Networks*, pp. 69–78, deu, September 2005.
 - [6] S. Deller, S. R. Bowling, and G. A. Rabadi, “Applying the Information Age Combat Model: Quantitative Analysis of Network Centric Operations,” *The International C2 Journal*, vol. 3, no. 1, pp. 1–25, 2009.
 - [7] B. G. Kang and T. G. Kim, “Reconfigurable C3 simulation framework: Interoperation between C2 and communication simulators,” in *Proceedings of the 2013 43rd Winter Simulation Conference - Simulation: Making Decisions in a Complex World, WSC 2013*, pp. 2819–2830, USA, December 2013.
 - [8] A. Tolk, *Engineering principles of combat modeling and distributed simulation*, John Wiley & Sons, 2012.
 - [9] HLA Working Group, IEEE standard for modeling and simulation (M&S) high level architecture (HLA)-framework and rules, IEEE Standard, 2000.
 - [10] U. S. DoD, TENA-the test and training enabling architecture, <https://www.tena-sda.org>.
 - [11] K. H. Lee, J. H. Hong, and T. G. Kim, “System of systems approach to formal modeling of CPS for simulation-based analysis,” *ETRI Journal*, vol. 37, no. 1, pp. 175–185, 2015.
 - [12] K.-M. Seo, K.-P. Park, and B.-J. Lee, “Achieving Data Interoperability of Communication Interfaces for Combat System Engineering,” *IEEE Access*, vol. 5, pp. 17938–17951, 2017.
 - [13] S. H. Choi, K.-M. Seo, and T. G. Kim, “Accelerated simulation of discrete event dynamic systems via a multi-fidelity modeling framework,” *Applied Sciences (Switzerland)*, vol. 7, no. 10, article no. 1056, 2017.
 - [14] K.-M. Seo, W. Hong, and T. G. Kim, “Enhancing model composability and reusability for entity-level combat simulation: A conceptual modeling approach,” *Simulation*, vol. 93, no. 10, pp. 825–840, 2017.
 - [15] C. Choi, K.-M. Seo, and T. G. Kim, “DEXSim: An experimental environment for distributed execution of replicated simulators using a concept of single simulation multiple scenarios,” *Simulation*, vol. 90, no. 4, pp. 355–376, 2014.
 - [16] B. P. Zeigler, T. G. Kim, and H. Praehofer, *Theory of Modeling and Simulation Integrating Discrete Event and Continuous Complex Dynamic Systems*, Academic Press, 2000.
 - [17] T. G. Kim, C. H. Sung, S.-Y. Hong et al., “DEVSIM++ Toolset for Defense Modeling and Simulation and Interoperation,” *The Journal of Defense Modeling and Simulation*, vol. 8, no. 3, pp. 129–142, 2011.
 - [18] T. R. Henderson, M. Lacage, G. F. Riley, C. Dowell, and J. Kopena, “Network simulations with the ns-3 simulator , in SIGCOMM demonstration,” *Network simulations with the ns-3 simulator , in SIGCOMM demonstration*, p. 527, August 2008.
 - [19] J. Pan and R. Jain, *A survey of network simulation tools: Current status and future developments*, Washington University, 2008.
 - [20] M.-W. Yoo, C. Choi, and T. G. Kim, “High-Level Architecture service management for the interoperation of federations,” *Simulation*, vol. 91, no. 6, pp. 566–590, 2015.
 - [21] D. S. Alberts, J. J. Garstka, and F. P. Stein, *Network Centric Warfare: Developing and Leveraging Information Superiority*, Assistant Secretary of Defense (C3I/Command Control Research Program), Washington DC, Wash, USA, 2000.
 - [22] S. K. Sukhpreet Kaur, “An Overview of Mobile Ad hoc Network: Application, Challenges and Comparison of Routing Protocols,” *IOSR Journal of Computer Engineering*, vol. 11, no. 5, pp. 7–11, 2013.
 - [23] M. F. Khan, E. A. Felemban, S. Qaisar, and S. Ali, “Performance analysis on packet delivery ratio and end-to-end delay of different network topologies in wireless sensor networks (WSNs),” in *Proceedings of the 9th IEEE International Conference on Mobile Ad-Hoc and Sensor Networks, MSN 2013*, pp. 324–329, China, December 2013.
 - [24] F. Gebali, “Modeling Network Traffic,” in *Analysis of Computer and Communication Networks*, Springer, 2008.
 - [25] F. Bai and A. Helmy, *A survey of mobility models , Wireless Adhoc Networks*, University of Southern, California, 2004.
 - [26] J. M. Choi and Y. B. Ko, “A performance evaluation for ad hoc routing protocols in realistic military scenarios,” in *Proceedings of the in Proceedings of the International Conference on Cellular and Intelligent Communications, CIC, 2004*, October 2004.
 - [27] C. Rajabhushanam and A. Kathirvel, “Survey of Wireless MANET Application in Battlefield Operations,” *International Journal of Advanced Computer Science and Applications*, vol. 2, no. 1, 2011.
 - [28] G. M. Patil, A. Kumar, and A. D. Shaligram, “Performance Analysis and Comparison of MANET Routing Protocols in Selected Traffic Patterns For Scalable Network,” vol. 6, pp. 109–117, 2016.
 - [29] S. Katiyar, R. Gujral, and B. Mallick, “Comparative performance analysis of MANET routing protocols in military operation using NS2,” in *Proceedings of the 1st International Conference on Green Computing and Internet of Things, ICGCIoT 2015*, pp. 603–609, India, October 2015.
 - [30] K. V. Vishwanath and A. Vahdat, “Realistic and responsive network traffic generation,” *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 4, pp. 111–112, 2006.
 - [31] F. Geyer, S. Schneelee, and G. Carle, “RENETO, a realistic network traffic generator for OMNeT++/INET,” in *Proceedings of the 6th International Conference on Simulation Tools and Techniques, SIMUTools 2013*, pp. 73–81, greece, March 2013.
 - [32] H. S. Bindra, S. K. Maakar, and A. L. Sangal, “Performance evaluation of two reactive routing protocols of MANET using group mobility model,” *International Journal of Computer Science*, vol. 7, no. 3, pp. 38–43, 2010.
 - [33] G. Kioumourtzis, C. Bouras, and A. Gkamas, “Performance evaluation of ad hoc routing protocols for military communications,” *International Journal of Network Management*, vol. 22, no. 3, pp. 216–234, 2012.
 - [34] A. Fongen, M. Gjellerud, and E. Winjum, “A military mobility model for MANET research,” in *Proceedings of the IASTED International Conference on Parallel and Distributed Computing and Networks, PDCN 2009*, pp. 213–219, aut, February 2009.
 - [35] S. Reidt and S. D. Wolthusen, “An evaluation of cluster head TA distribution mechanisms in tactical MANET environments,” in *Proceedings of the in Proceedings of the International Technical Alliance in Network and Information Science, NIS-ITA, 2007*.
 - [36] X. Lu, Y. C. Chen, I. Leung, Z. Xiong, P. Liò et al., “A novel mobility model from a heterogeneous military MANET trace,” in *in Proceedings of the International Conference on Ad-Hoc*

- Networks and Wireless, ADHOC-NOW*, pp. 463–474, France, September 2008.
- [37] H. Seo, S. H. Kim, and J. S. Ma, “A novel mobility model for the military operations with real traces,” in *Proceedings of the in Proceedings of the International Conference on the Advanced Communication Technology*, pp. 129–133, Korea, February 2010.
- [38] C. M. Seo, B. P. Zeigler, D. H. Kim, and K. Duncan, “Integrating web-based simulation on IT systems with finite probabilistic DEVS,” in *Proceedings of the in Proceedings of the Symposium on Theory of Modeling Simulation: DEVS Integrative MS Symposium*, pp. 173–180, USA, April 2015.
- [39] Y. Kodratoff, *Introduction to machine learning*, Morgan Kaufmann, 2014.
- [40] Z. Xiao, L. Peng, Y. Chen, H. Liu, J. Wang, and Y. Nie, “The Dissolved Oxygen Prediction Method Based on Neural Network,” *Complexity*, vol. 2017, Article ID 4967870, pp. 1–6, 2017.
- [41] T. Gligorijević, Z. Ševarac, B. Milovanović et al., “Follow-up and risk assessment in patients with myocardial infarction using artificial neural networks,” *Complexity*, vol. 2017, Article ID 8953083, 8 pages, 2017.
- [42] G. Neuneck, “The revolution in military affairs: Its driving forces, elements, and complexity,” *Complexity*, vol. 14, no. 1, pp. 50–61, 2008.
- [43] C. E. Maldonado and N. A. Gómez Cruz, “The complexification of engineering,” *Complexity*, vol. 17, no. 4, pp. 8–15, 2012.
- [44] P. Kuosmanen, *Choosing routing protocol for military ad hoc networks based on network structure and dynamics*, Helsinki University of Technology, 2002.
- [45] B. G. Kang, B. S. Kim, and T. G. Kim, “Abstraction on network model under interoperable simulation environment,” in *Proceedings of the 30th European Conference on Modelling and Simulation, ECMS 2016*, pp. 460–466, DEU, June 2016.
- [46] A. H. Dekker, *C4ISR architectures, social network analysis and the FINC methodology: an experiment in military organisational structure*, Defence Science and Technology Organisation, 2002.
- [47] I. C. Moon, K. M. Carley, and T. G. Kim, *Modeling and Simulating Command and Control: For Organizations Under Extreme Situations*, Springer Science & Business Media, 2013.
- [48] I. Porche, R. Isaac, and W. Bradley, “The impact of network performance on warfighter effectiveness,” *RAND ARROYO CENTER SANTA MONICA CA*, 2006.
- [49] I. Porche, L. Jamison, and T. Herbert, *Framework for Measuring the Impact of C4ISR Technologies and Concepts on Warfighter Effectiveness Using High Resolution Simulation*, Rand Arroyo Center Santa Monica CA, 2004.
- [50] N. E. Miner, B. P. Van Leeuwen, K. M. Welch et al., “Evaluating communications system performance effects at a system of systems level,” in *Proceedings of the 2012 IEEE Military Communications Conference, MILCOM 2012*, USA, November 2012.
- [51] S. M. Sanchez and H. Wan, “Better than a petaflop: The power of efficient experimental design,” in *Proceedings of the 2009 Winter Simulation Conference, WSC 2009*, pp. 60–74, USA, December 2009.
- [52] C. K. McCalley, B. J. Woodcroft, S. B. Hodgkins et al., “Methane dynamics regulated by microbial community response to permafrost thaw,” *Nature*, vol. 514, no. 7253, pp. 478–481, 2014.
- [53] J. H. Lee, H. S. Tag, G. T. Kim et al., “Effect of Rheumatoid Factor on Vascular Stiffness in General Population without Joint Symptoms,” *Kosin Medical Journal*, vol. 32, no. 1, p. 25, 2017.




Hindawi

Submit your manuscripts at
www.hindawi.com

