

분산 컴퓨팅 환경에서의 워게임 시물레이션을 위한 네트워크 트래픽 제어

장성호¹ · 김태영¹ · 이종식^{1*}

Network Traffic Control for War-game Simulation in Distributed Computing Environment

Sung Ho Jang · Tae Young Kim · Jong Sik Lee

ABSTRACT

The distributed war-game simulation system has been used to represent the virtual battlefield environment. In order to produce a simulation result, simulators connected from a network transfer messages with location information of simulated objects to a central simulation server. This network traffic is an immediate cause of system performance degradation. Therefore, the paper proposes a system to manage and control network traffic generated from distributed war-game simulation. The proposed system determines the moving distance of simulated objects and filters location messages by a distance threshold which is controlled according to system conditions like network traffic and location error. And, the system predicts the next location of simulated objects to minimize location error caused by message filtering. Experimental results demonstrate that the proposed system is effective to control the network traffic of distributed war-game simulation systems and reduce the location error of simulated objects.

Key words : War game, Network traffic control, Message filtering, Distributed simulation, Location prediction

요약

현재 복잡한 가상전장 환경을 표현하기 위해 많은 수의 시물레이터들을 연결한 분산 컴퓨팅 기반의 워게임 시물레이션이 이용되고 있다. 모의 결과를 도출하기 위해 네트워크상의 각 시물레이터들은 중앙 모의 서버에 주기적으로 메시지를 송수신하게 된다. 이러한 네트워크 트래픽은 전체 시스템 성능저하의 직접적인 원인이 된다. 따라서 본 논문에서는 분산 워게임 시물레이션 시 발생하는 네트워크 트래픽을 제어하기 위한 시스템 및 방법을 제안한다. 제안된 시스템은 메시지 수신 시 해당 모의 개체의 이동거리를 측정하고 이를 거리 임계값과 비교하여 해당 메시지를 필터링한다. 거리 임계값은 네트워크 트래픽 및 위치 오차와 같은 시스템 조건에 따라 조절된다. 그리고 모의 개체의 위치를 예측함으로써 메시지 필터링 시 발생하는 위치오차를 최소화한다. 실험 결과는 제안된 시스템이 분산 시물레이션 시스템의 네트워크 트래픽을 감소시키고 메시지 필터링에 따른 모의 개체의 위치 오차를 줄이는 데 매우 효과적이라는 것을 입증한다.

주요어 : 워게임, 네트워크 트래픽 제어, 메시지 필터링, 분산 시물레이션, 위치 예측

1. 서론

현재 미국, 일본 등과 같은 선진국을 주도로 다양한 워

게임 시물레이션 모델이 개발되어 군사 훈련, 무기 개발, 전투 실험 등과 같은 국방 관련 주요 분야에 적용되고 있다^{1,2)}. 초창기 워게임 모델은 싱글 머신을 기반으로 한 단순 시물레이션 시스템이었다. 그러나 네트워크 및 IT 기술의 발달과 함께 점차 무기체계 M&S 개발 환경이 복잡하고 특수한 기능의 수행을 필요로 함에 따라 상호 연동성, 상호 운용성 및 재사용성을 갖춘 다양한 분산 워게임 시물레이션 모델들이 연구 및 개발되었다³⁾. 분산 워게임 시물레이션 모델은 복잡한 가상 전장 환경을 표현하기 위해 분산 네트워크상에서 많은 수의 이기종 시물레이터들

* 본 연구는 방위사업청과 국방과학연구소의 지원으로 수행되었습니다.(UD080042AD)

2009년 5월 18일 접수, 2009년 10월 13일 채택

¹⁾ 인하대학교 정보공학과

주 저 자 : 장성호

교신저자 : 이종식

E-mail: ho7809@hanmail.net

을 연결하고 이들 간의 상호작용을 통해 무기체계 소요분석, 개발, 시험평가, 훈련 등과 같은 군사용 시뮬레이션 및 모델링을 실현한다⁴⁾.

분산 컴퓨팅 환경 하에서의 위게임 시뮬레이션은 가상 전장 환경의 각 모의 개체를 표현하는 시뮬레이터들과 중앙 모의 서버간의 주기적이고 반복적인 메시지 통신을 통해 이루어진다. 또한, 모의 개체의 이동에 따라 해당 개체의 위치 정보 역시 이러한 메시지 통신을 통해 전달되며 갱신되는데 이는 전체 시스템 성능에 아주 많은 영향을 미친다. 즉, 네트워크 트래픽의 증가는 통신 회선의 점유 시간을 늘리고 중앙 서버의 작업 부하를 높임으로써 시스템 성능 저하의 원인이 된다. 이는 전송 오류를 증가시키고 동시에 데이터 손실을 발생시킬 수 있으므로 네트워크 트래픽 제어는 분산 시뮬레이션의 중요한 화두라고 할 수 있다. 그러나 현재까지 개발된 위게임 시뮬레이션 모델들의 대부분이 가상 전장 환경의 구현 및 전투모의 구성에만 초점을 맞추고 있는 실정이다.

따라서 본 논문은 분산 위게임 시뮬레이션 시 각 시뮬레이터가 발생시키는 모의 개체의 위치 정보 메시지 수를 감소시키고 동시에 안정적인 모의 결과를 제공하기 위하여 이동성을 갖는 모의 개체의 위치를 예측하는 네트워크 트래픽 시스템 및 방법을 제안한다. 제안된 시스템은 모의 개체의 이동거리를 측정하고 특정 거리 임계값을 기준으로 이동성에 따른 위치정보 메시지를 필터링하여 분산 시뮬레이션 시스템의 작업 부하 및 네트워크 트래픽을 감소시킨다. 또한 메시지 필터링에 따른 위치 오차를 줄이기 위해 통계적 예측 기법을 이용하여 모의 개체의 위치를 예측한다.

본 논문의 구성은 다음과 같다. 2장에서는 분산 컴퓨팅 환경 기반의 위게임 시뮬레이션에 대하여 설명하고 3장에서는 제안된 시스템의 구성 및 알고리즘에 대해 알아본다. 4장에서는 시뮬레이션 결과를 통해 제안된 시스템의 성능을 평가하고 마지막으로 5장에서 결론을 맺는다.

2. 관련 연구

2.1 분산 컴퓨팅 환경에서의 위게임 시뮬레이션

이장에서는 분산 컴퓨팅 환경 기반의 위게임 시뮬레이션에 대하여 소개하고 이에 대한 문제점 및 해결 방안을 도출한다. 앞 장에서 언급한 바와 같이 위게임 시뮬레이션은 네트워크를 이용한 분산 컴퓨팅 환경 기반으로 점차 발전되어 왔다. 세계 최초의 네트워크 기반 위게임 시뮬레이션 체계인 SIMNET(SIMulation NETwork)⁵⁾를 시

작으로 대규모 가상 환경에서 다수의 시뮬레이터들 간의 상호작용을 통해 대화형 분산 시뮬레이션을 시도한 DIS (Distributed Interactive Simulation)⁶⁾, 개략 모사 수준의 전투 객체를 모델링하고 논리적 시간을 기반으로 분산 시뮬레이션을 실시한 ALSP(Aggregate Level Simulation Protocol)⁷⁾와 같은 다양한 분산 위게임 시뮬레이션 모델 및 기법들이 개발되어 왔다. 또한, 현재에는 이기종 시뮬레이터 간의 상호 운용성 및 컴포넌트 재사용성을 고려한 분산 실시간 시뮬레이션 아키텍처인 HLA/ RTI(High Level Architecture / Run Time Infrastructure)⁸⁾가 국제 표준으로 인증 받아 대규모 전투모의 훈련 및 무기 체계 분석과 같은 다양한 국방 관련 시뮬레이션에 활용되고 있다. 이와 같은 대부분의 분산 컴퓨팅 환경 기반의 위게임 시뮬레이션 모델들은 기본적으로 그림 1과 같은 구조로 구성되어 있다.

분산 컴퓨팅 환경에서 각 시뮬레이터(Simulator)들은 전장 환경 내에서 모의되는 개체(Object)를 생성하고 이를 시뮬레이션하며 이에 대한 데이터 및 정보를 중앙 모의 서버(Central Simulation Server)에 전송한다. 또한 원활한 위게임 시뮬레이션 진행을 위해 가상 전장 상에서 이동하는 모의 개체를 담당하는 각 시뮬레이터는 중앙 모의 서버와 일정 시간 마다 주기적인 메시지 통신을 통해 해당 개체의 위치 정보를 갱신하게 된다. 이러한 주기적인 메시지 송수신은 전체 시스템의 성능과 밀접한 관계를 가진다. 제한된 성능의 컴퓨팅 및 네트워크 자원을 이용하는 분산 컴퓨팅의 특성 상 많은 수의 통신 메시지는 중앙 모의 서버의 작업 부하를 급격히 증가시키며 네트워크 병목현상을 일으킬 수 있다. 따라서 분산 시뮬레이션 시

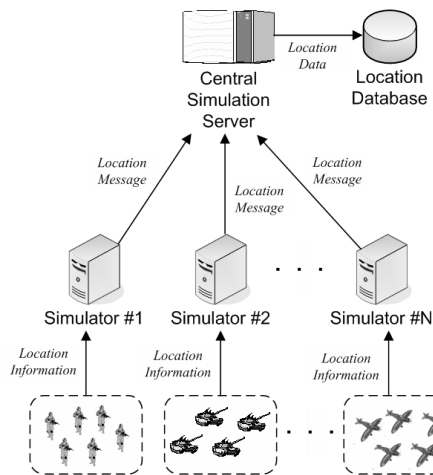


그림 1. 분산 컴퓨팅 환경 기반의 위게임 시뮬레이션

시스템 상에서 반복적으로 발생하는 네트워크 트래픽의 제어 기법이 필요하다. 그러나 대부분의 분산 워게임 시뮬레이션 모델들은 모의 객체 모델링, 객체 간 동기화 및 시간 관리, 모의 결과 분석 등에 초점을 맞추고 있으며 네트워크 트래픽 제어에 관한 연구는 아주 미비한 실정이다.

본 논문에서는 네트워크 트래픽 제어를 위한 방법으로 모의 개체의 이동거리에 따른 메시지 필터링 및 이동체 위치 예측을 제안한다. 위치 정보 갱신을 위한 메시지가 시뮬레이터로부터 전송되기 전 메시지 내의 이동 정보를 바탕으로 메시지 필터링을 실시하여 네트워크 트래픽을 효과적으로 감소시키며 예측 알고리즘을 이용하여 메시지 필터링 시 발생하는 모의 개체의 위치 오차를 줄인다.

2.2 분산 환경에서의 네트워크 트래픽 제어

앞서 설명한 바와 같이 분산 컴퓨팅 환경에서의 워게임 시뮬레이션에는 네트워크 트래픽 제어가 필수적이다. 현재까지 분산 컴퓨팅 환경에서의 네트워크 트래픽을 관리하기 위해 다양한 모델들이 제안되었다.

대표적인 모델로는 분산 시스템에서 다양한 상호작용을 위한 메시지를 하나의 패키지 형태로 목적 호스트에 전송하는 모바일 에이전트 모델⁹⁾, 분산 시뮬레이션 환경인 DIS(Distributed Interactive Simulation)에서 발생하는 불필요한 메시지 트래픽을 필터링하는 그리드 기반의 데이터 분산 관리 모델¹⁰⁾, DIS에서 네트워크 트래픽을 관리하기 위한 네트워크 트래픽 예측 모델¹¹⁾ 등이 있다.

그러나 이러한 모델들은 워게임 시나리오, 모의 전투 개체 등과 같은 분산 워게임 시뮬레이션의 특징을 전혀 고려하지 않고 있다. 따라서 우리는 워게임 시나리오에 따른 모의 개체의 이동성을 고려하여 유연한 메시지 필터

링이 가능한 네트워크 트래픽 제어 시스템을 제안한다.

3. 분산 워게임 시뮬레이션을 위한 네트워크 트래픽 제어 시스템

이 장에서는 분산 워게임 시뮬레이션 환경 상에서 각 시뮬레이터로부터의 위치 정보 메시지를 필터링하고 이때 발생하는 위치오차를 줄이기 위해 이동 개체의 위치 예측을 실시하는 네트워크 트래픽 제어 시스템 및 방법을 설명한다. 본 논문에서 제안한 네트워크 트래픽 제어 시스템의 구성은 그림 2와 같다.

기존 분산 워게임 시뮬레이션과 달리 제안된 시스템은 메시지 필터(Filter)와 위치 예측 모듈(Location Predictor)을 포함한다. 2장에서 설명한 바와 같이 분산된 시뮬레이터들은 보병, 전차, 전투기 등과 같은 다양한 모의 개체를 생성하고 각 모의 개체가 가상 전장 환경 상에서 이동할 경우 이에 대한 이동정보를 중앙 모의 서버에 전송하게 된다. 이 때, 메시지 필터는 분산된 시뮬레이터로부터 모의 개체의 위치 정보 메시지가 수신될 경우 이를 특정 거리 임계값에 따라 필터링함으로써 이동 개체의 위치 갱신 여부 및 메시지 전송을 제어하는 역할을 담당한다. 위치 예측 모듈은 메시지 필터링 시 발생하는 모의 개체의 위치 오차를 줄이기 위해 위치 데이터베이스(Location Database)에 저장된 각 이동개체의 이동 경로(Historical Data)를 바탕으로 각 모의 개체의 다음 번 이동 위치를 예측함으로써 메시지 필터링에 따른 위치 오차를 최소화하게 된다.

분산 워게임 시뮬레이션 시스템의 네트워크 트래픽 제어 시 가장 중요한 부분은 모의 개체 위치의 갱신 주기이다. 너무 빈번한 위치 정보 갱신은 통신 회선 점유 및 통

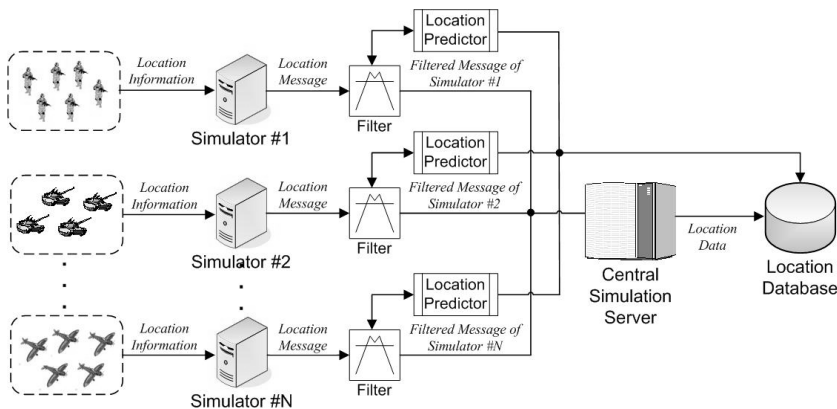


그림 2. 분산 워게임 시뮬레이션을 위한 네트워크 트래픽 제어 시스템의 구성도

```
// Md = Moving Distance of Simulated Object
// Th = Distance Threshold
Simulator inputs a location message to Filter
if Md <= Th
    Message is filtered
else
    Message is sent to Central Simulation Server
end if
```

그림 3. 이동 거리 기반의 메시지 필터링

```
// Le = Average Location Error of Mobile Object
// Th = Distance Threshold (default value = High Threshold)
// MaxMD = Maximum Moving Distance of Mobile Object
if Le >= 20 % of MaxMD
    Th = Low Threshold
else
    Th = High Threshold
```

그림 4. 거리 임계값 조절

신 부하 증가로 인해 전체 시스템 성능 저하의 주요 원인으로 작용한다. 또한, 너무 드문 위치 정보 갱신은 각 모의 개체의 위치 오차를 증가시켜 전투 모의와 같은 분산협동 처리 시 개체 간 충돌 및 데이터 오류를 발생시킨다. 따라서 어떠한 기준으로 모의 개체의 위치를 갱신하고 메시지를 필터링할 것인가는 네트워크 트래픽 제어의 핵심 요소이다. 본 논문에서는 기본적인 필터링 기준 요소인 위치 메시지의 전송 시간 간격 대신 모의 개체의 이동 거리를 적용한다. 모의 개체의 이동 속도는 시간에 따라 일정하지 않고 동적으로 변화하기 때문이다. 또한 비교적 짧은 시간동안 모의 개체가 먼 거리를 이동할 경우, 상당한 위치 오차가 발생할 수 있다. 그러므로 일정 시간 간격에 따라 모의 개체의 위치를 갱신하는 것보다 모의 개체의 이동 거리에 따라 위치를 갱신하는 것이 동적인 이동성을 갖는 모의 개체의 메시지 필터링에 더 효과적이다.

그림 3은 모의 개체의 이동성에 따른 메시지 필터링 과정이다. 메시지 필터는 시뮬레이터로부터 위치 정보 메시지를 전달받을 경우 메시지 안에 포함된 모의 개체의 이동 거리(Moving Distance)와 거리 임계값(Distance Threshold)을 비교한다. 모의 개체의 이동 거리는 다음과 같이 측정된다. 각 개체는 x, y, z 좌표를 이용하여 자신의 위치 p 를 표현한다. 개체의 이동 전 좌표를 $p_1(x_1, y_1, z_1)$ 이라고 하고 개체의 이동 후 좌표를 $p_2(x_2, y_2, z_2)$ 라고 가정하였을 경우 일정 시간 t 동안 모의 개체의 이동 거리 $M_{p_1 \rightarrow p_2}(t)$

는 피타고라스의 정리^[21]를 이용하여 아래의 식 (1)과 같이 계산할 수 있다.

$$M_{p_1 \rightarrow p_2}(t) = \frac{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}}{\quad} \quad (1)$$

이와 같이 계산된 모의 개체의 이동 거리가 거리 임계값 이하일 경우 메시지를 필터링하고 그렇지 않을 경우 해당 메시지를 중앙 모의 서버에 전송한다. 본 논문에서는 두 가지 타입의 거리 임계값(High Threshold, Low Threshold)을 이용한다. 두 거리 임계값의 실제적인 수치는 이동개체의 타입 및 평균 이동거리등을 이용하여 결정된다. 우리는 메시지 전송량과 위치 오차의 트레이드오프 관계를 통해 상황에 따라 거리 임계값을 조절한다.

그림 4는 시스템 상황에 따른 거리 임계값의 조절 과정을 보여준다. 트래픽 최소화를 목표로 하는 메시지 필터의 특성 상 기본적인 거리 임계값은 High Threshold이다. 필터링 후 메시지 필터는 모의개체의 평균 예측 오차가 해당 모의 개체의 최대 이동 거리의 20% 이상으로 증가할 경우 거리 임계값을 Low Threshold로 변경한다.

이와 같이 임계값 조절을 통해 위치 오차가 어느 수준 이상 늘어나는 것을 방지할 수 있지만 본 논문에서는 일정 수준의 네트워크 트래픽을 유지하면서 위치 오차를 최소화할 방법으로 모의 개체의 위치 예측을 적용한다. 물론 모의 개체의 이동성(Mobility)을 완벽히 예측한다는 것은 현실적으로 거의 불가능하다. 그러나 위게임 시뮬레이션상의 모의 개체들은 정해진 시뮬레이션 시나리오에 의해 이동성을 갖게 된다. 따라서 시뮬레이션 시나리오에 따라 객체들의 이동 경로에 대한 정보를 알고 있을 경우 충분히 모의 개체의 위치를 예측할 수 있다.

그림 5와 같은 산악지형에서 3개의 소대 A, B, C가 진지를 구축하고 정찰을 나간다고 가정하자. 모든 소대원은 A2지역에서 출발하여 도보로 F2지역까지 이동을 한 다음 진지를 구축한다. 그 다음, 각자 맡은 지시에 따라 A소대는 진지 방어를 위한 주변 경계근무에 돌입하고 B소대는 아래의 산길을 따라 C소대는 오른쪽의 대로를 따라 수색작업에 나선다. 이러한 시나리오에 따라 A소대, B소대, C소대의 이동경로는 다음과 같이 예상할 수 있다.

- A소대 = A2→B2→C2→D2→E2→F2→{E1, E2, F1, F2}
- B소대 = A2→B2→C2→D2→E2→F2→G2→G3→G4→G5→H5→H6→G6→H6→H7→H8→I8

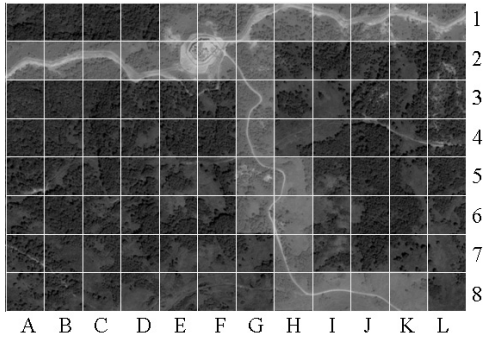


그림 5. 모의 개체의 이동 시나리오 예시

→J8→K8

- C소대 = A2→B2→C2→D2→E2→F2→G3→H3
→I3→J3→K3→L3

앞서 말한 바와 같이 모의 개체의 완벽한 좌표표를 예측할 순 없지만 이러한 시뮬레이션 시나리오를 바탕으로 예측 알고리즘과 현재 지역에 따른 이동 지역의 제약조건을 이용할 경우 어느 정도 정확한 예측이 가능하다. 예를 들어, 모의 개체 1이 A소대에 속하는 소대원이며 현재 위치 p_1 이 F2 지역에 포함될 경우, 모의 개체 1의 다음 위치 p_2 는 아래와 같이 제한된다.

$$p_2 \in E1, E2, F1, F2 \text{ (if } p_1 \in F2)$$

본 논문에서는 대표적인 예측 알고리즘인 이중 지수 평활법(Double Exponential Smoothing)^[13]을 응용하여 식 (2)와 같이 모의개체의 이동거리를 추정한다. 이중 지수 평활법은 최근 관측 자료에 큰 가중치를 두고 과거 자료일수록 작은 가중치를 두는 일종의 가중평균 기반의 예측방법으로 직관적인 예측이 가능하고 많은 자료의 예측 시 편리하다는 장점을 갖는다. 우리는 이중 지수 평활법을 적용함으로써 위치 이동이 빈번하게 일어나고 시나리오에 따른 일정한 이동 패턴, 즉 추세를 가진 워게임 모의 개체의 특성을 고려하여 이동거리를 예측한다.

$$Pd_{t+1} = \prod_{i=1}^n [(1-\alpha)A_t + \alpha B_t] \times Pd_t \quad (2)$$

where $A_t = Pd_t/Pd_{t-1}$
 $B_t = Md_t/Md_{t-1}$

식 (2)에서 Pd 는 모의개체의 예측된 이동거리, Md 는 실제 측정된 이동거리이다. 여기서 이동거리는 모의 개체의 x 방향 이동거리 또는 y 방향 이동거리를 의미한다. Pd_t

는 모의개체의 기본적인 이동거리이며 a 는 예측 이동거리와 실제 이동거리간의 가중치이다. 본 논문에서는 실험을 통해 최적의 a 값으로 0.4를 선택하였다. 이는 4장의 실험결과를 통해 입증하도록 한다. 이렇게 추정된 x 방향 및 y 방향 이동거리인 xPd_{t+1}, yPd_{t+1} 를 이용하여 모의 개체의 다음번 이동 위치 (x_{t+1}, y_{t+1}) 를 아래의 식 (3), (4)와 같이 예측한다.

$$x_{t+1} = x_t + xPd_{t+1} \quad (3)$$

$$y_{t+1} = y_t + yPd_{t+1} \quad (4)$$

z 좌표는 위치 정보 데이터베이스에 저장된 전장 지도 정보를 이용하여 (x_{t+1}, y_{t+1}) 에 해당하는 지역의 고도(altitude)로 대체한다. 위치 정보 데이터베이스에 저장된 전장 지도 정보는 시뮬레이션 시나리오와 함께 이동 제약조건의 설정에 이용된다.

위치 예측 모듈은 이렇게 예측된 모의 개체의 위치 정보를 위치 정보 데이터베이스에 전송하여 메시지 필터링 시 해당 모의 개체의 위치 정보로 대체함으로써 전투모의 시 발생할 수 있는 위치 오차로 인한 개체 충돌, 데이터 손실 및 오류 등을 줄여줄 수 있다.

4. 실험 결과

본 논문에서는 성능 평가를 위해 DEVS 시뮬레이션 & 모델링 환경^[14] 상에서 제안된 위치 예측 기반의 네트워크 트래픽 제어 시스템을 다양하게 시뮬레이션하고 성능 지표를 측정하였다. 실험에서는 전장 환경 상의 이동 객체의 타입을 보병으로 한정하여 그림 5의 시나리오를 바탕으로 시뮬레이션 하였다. 실제 모의 전장 환경 상에서는 보병, 전차, 차량, 헬기, 전투기 등 수많은 타입의 모의 개체가 존재하지만 실험의 복잡성을 간소화하고 이동 패턴의 추출을 손쉽게 하기 위해 가장 기본적인 전투 요소인 보병을 모델로 선택하였다. 전체 전장은 그림 5와 같이 100×100의 크기(Pixel)를 가진 96개의 지역(area)으로 구성되었으며 Random Way point 모델^[15]을 응용하여 식 (5), 식 (6)과 같이 각 모의 개체의 x 좌표와 y 좌표를 일정 시간간격마다 변화시켰다. z 좌표의 변화는 3장에서 설명한 것과 같이 위치 정보 데이터베이스에 저장된 해당 x, y 좌표의 고도를 고려하였다. 식 (5)와 (6)에서 \vec{s} 는 모의 개체의 기본 이동속도를 나타내고 Θ 는 속도 편차율을 나타내며 S_{max} 는 최고 속도 편차를 나타낸다. $random()$ 은 -1부터 1까지의 값을 반환하는 랜덤 함수를 의미한다.

$$x_{t+1} = x_t + (\vec{s} + random()) \times \Theta \times s_{max} \quad (5)$$

$$y_{t+1} = y_t + (\vec{s} + random()) \times \Theta \times s_{max} \quad (6)$$

제안된 시스템의 성능을 평가하기에 앞서 시스템에서 사용된 예측모델의 예측 정확도를 측정하였다. 우리는 예측 정확도를 나타내는 성능지표로서 식 (7)과 같이 절대평균 백분비 오차(MAPE: Mean Absolute Percentage Error)^[16]를 선택하여 반복적인 실험을 통해 예측 결과를 도출하였다. 또한, 예측 정확도의 개선을 위한 최적의 α 값을 찾기 위해 α 값을 다양하게 변화시켜 각 예측 데이터의 절대평균 백분비 오차를 측정하였다. 식 (7)에서 Y_t 는 실제 이동 거리를 F_t 는 예측 이동거리를 가리키며 n 은 이동거리 데이터의 수를 의미한다.

$$MAPE = (\sum(Y_t - F_t) / Y_t) \times 100 / n \quad (7)$$

표 1은 α 값의 변화에 따른 절대평균 백분비 오차를 보여준다. 절대평균 백분비 오차가 0에 가까울수록 더 높은 예측 정확도를 제공하는 것을 의미하고 10%이하 일 경우 아주 정확한 예측 알고리즘이라고 말할 수 있다. 표 1의 결과에 따라 α 값이 0.4일 경우 가장 낮은 절대평균 백분비 오차를 제공하는 것을 알 수 있다. 따라서 우리는 α 값이 0.4로 고정된 후 제안된 네트워크 트래픽 제어 시스템의 성능 평가를 실시하였다.

그림 6은 중앙 모의 서버가 각 분산 시뮬레이터로부터 전달받은 총 위치 정보 메시지의 수를 보여준다. 시뮬레이션 시간 동안 분산 시뮬레이터들이 발생시킨 메시지의 수는 총 12000개이다. 우리는 메시지 필터링을 위해 거리 임계값을 세 가지 형태(Low Threshold, High Threshold, Threshold Tradeoff)로 조절하여 메시지 수를 측정하였다. Low Threshold의 경우, 3618개의 메시지를 필터링하여 30.15%의 메시지 감소율을 기록하였다. 이에 반해 High Threshold의 경우는 5732개의 메시지를 필터링하여 가장 높은 47.77%의 메시지 감소율을 기록하였다. 또한 High Threshold와 Low Threshold를 혼재하여 사용하는 Threshold Tradeoff의 경우 5264개의 메시지를 필터링하여 High Threshold의 메시지 감소율과 거의 유사한 43.87%의 메시지 감소율을 기록하였다.

또한 논문에서 제안된 위치 예측기법의 성능 평가를 위해 우리는 그림 7과 같이 모의 개체의 위치 예측에 따른 각 모의개체의 평균 위치 오차를 측정하였다. 평균 위치 오차 e 는 식 (8)과 같이 계산된다. 식 (8)에서 n 은 모의

표 1. α 값의 변화에 따른 절대평균 백분비오차

Alpha (α) 값	절대평균 백분비오차 (%)
0.1	11.53673
0.2	9.142434
0.3	8.906877
0.4	8.760155
0.5	8.882692
0.6	10.91756
0.7	12.93388
0.8	15.11017
0.9	17.71788

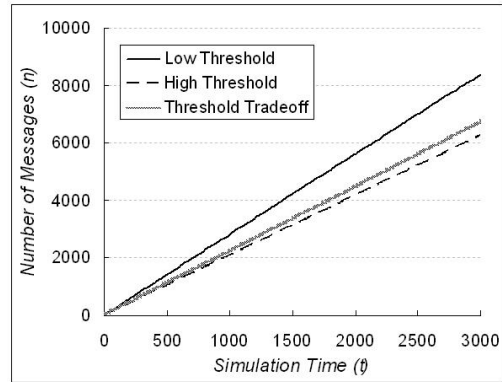


그림 6. 거리 임계값에 따른 통신 메시지 수 비교

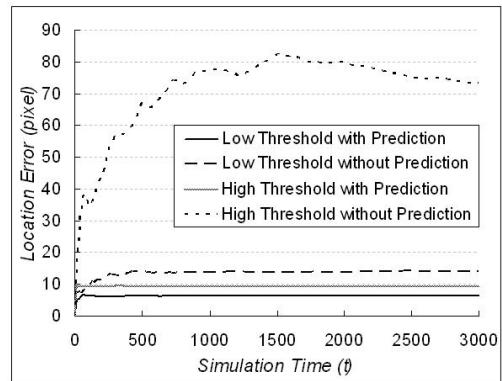


그림 7. 모의 개체의 위치 예측에 따른 위치 오차 비교

개체의 수를 나타내며 m 은 위치 데이터의 수를 나타낸다. ls 는 위치 정보 데이터베이스에 저장된 모의개체의 위치를 나타내며 lr 은 실제 모의개체의 위치를 나타낸다.

$$e = \frac{1}{n \times m} \sum_{i=1}^n \sum_{j=1}^m |ls_j - lr_j| \quad (8)$$

그림 7의 시뮬레이션 결과에서 볼 수 있듯이 Low Threshold의 경우 위치 예측 기법 적용 시 평균 위치 오차는 6.06이었으며 그렇지 않은 경우에는 13.42이었다. 이는 위치 예측 기법의 적용을 통해 54%의 위치 오차를 줄일 수 있다는 것을 보여준다. 네트워크 트래픽 감소를 위해 잦은 필터링이 발생하는 High Threshold의 경우 위치 예측 기법 적용 시 평균 위치 오차는 9.23이었으며 그렇지 않은 경우에는 71.49이었다. 이는 87%의 위치 오차 감소율로 잦은 필터링으로 인해 위치 갱신이 이루어지지 않는다 하더라도 위치 예측을 통해 위치 오차를 확연히 줄여 준다는 것을 증명한다.

그림 7의 결과를 통해 Low Threshold를 기준으로 메시지 필터링을 실시하였을 경우 High Threshold보다 34% 감소된 평균 위치 오차를 제공했다는 사실을 알 수 있다. 그러나 이 결과를 바탕으로 Low Threshold가 High Threshold보다 우수하다고 판단하기는 어렵다. 그림 6의 결과와 같이 High Threshold는 Low Threshold보다 25% 적은 메시지 수를 기록하였기 때문이다. 이처럼 Low Threshold와 High Threshold는 각각 위치 오차의 최소화, 네트워크 트래픽의 최소화에 초점을 맞추고 있다. 우리는 이러한 거리 임계값들의 특징을 고려하여 Threshold Tradeoff 방식을 제안하였으며 그림 6의 결과를 통해 High Threshold와 유사하게 위치 정보 메시지 수를 줄일 수 있다는 것을 증명하였다. 그리고 얼마만큼의 위치 오차를 줄일 수 있는 지 알아보기 위해 아래의 그림 8과 같이 평균 위치오차를 측정하였다.

그림 8에서 볼 수 있듯이 시뮬레이션 초기에는 High

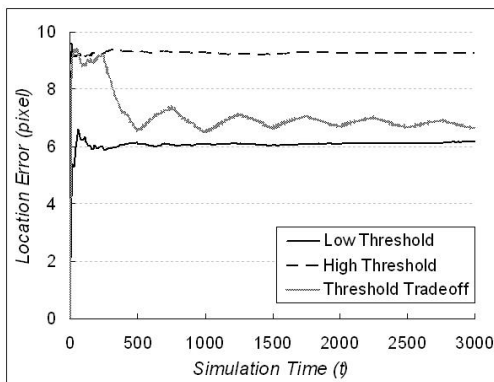


그림 8. 거리 임계값에 따른 위치 오차 비교

Threshold와 유사하게 평균 위치 오차 10을 기록했으나 거리 임계값 조정을 통해 점차 상승 하강을 반복하며 최종적으로는 Low Threshold와 유사한 6.63의 평균 위치 오차를 기록하였다. 이와 같은 결과는 Threshold Tradeoff를 제안된 시스템에 적용할 경우 Low Threshold와 High Threshold의 목표인 위치 오차와 네트워크 트래픽의 최소화를 동시에 실현한다는 것을 보여준다.

5. 결 론

본 논문에서 우리는 분산 워게임 시뮬레이션 시 발생하는 네트워크 트래픽을 제어하기 위한 시스템 및 방법을 제안하였다. 제안된 시스템은 거리 임계값을 기준으로 모의 개체의 이동거리에 따라 필터링을 실시하는 메시지 필터와 메시지 필터링 시 발생하는 위치오차를 최소화하기 위해 모의 개체의 이동거리를 추정하여 다음 번 위치를 예측하는 위치 예측 모듈로 구성된다. 또한 논문에서는 거리 임계값을 효과적으로 조절하기 위해 Threshold Tradeoff 방식을 시스템에 적용하였다. 실험결과를 통해 논문에서 제안된 Tradeoff 기반의 메시지 필터링과 위치 예측 기법이 분산 시뮬레이션 시스템의 네트워크 트래픽을 감소시키고 메시지 필터링 시 발생하는 위치 오차를 줄이는 데 매우 효과적이라는 것을 증명하였다. 그러나 이는 제한된 환경 상에서 한 가지 타입의 모의 개체를 기준으로 시뮬레이션을 실시한 결과이므로 향후 연구에서는 대규모 분산 네트워크 시뮬레이션 환경인 HLA/RTI^[8]상에서 다양한 시나리오를 바탕으로 여러 가지 타입의 모의 개체들에 대한 모델링 및 시뮬레이션을 진행하고자 한다.

참 고 문 헌

1. 김진우 (2005), “유럽 국가들의 국방 시뮬레이션 동향과 시사점”, *주간국방논단*, 제1065호, pp. 1-10.
2. 유병주, 이종호, 권오정 (2005), “국방 M&S를 활용한 훈련 체계 혁신방안”, *국방정책연구*, 제68호, 한국국방연구원, pp. 67-93.
3. 이종호 (2008), *모델링 및 시뮬레이션 - 이론과 실제*, 이십일세기군사연구소.
4. 이규노, 왕지현 (2008), “지상무기체계 전투효과 분석을 위한 통합 시뮬레이션”, *국방과학기술 플러스*, 제64호, pp. 1-12.
5. Miller, D. and Thorpe, J. A. (1995), “SIMNET : The advantage of simulator networking”, *Proceedings of the IEEE*, Vol. 83, Issue 8, pp. 1114-1123.

6. Fullford, D. (1996), "Distributed interactive simulation: itaposs; past, present, and future", *Proc. of the Winter Simulation Conference 1996*, California, pp. 179-185.
7. Weatherly, R.M., Wilson, A.L., and Griffin, S.P. (1993), "ALSP - Theory, Experience, and Future Directions", *Proc. of the Winter Simulation Conference 1993*, Los Angeles,, pp. 1068-1072.
8. Shi X. and Zhong Q. (2003), "The introduction on high level architecture (HLA) and run-time infrastructure (RTI)", *Proc. of the SICE 2003 Annual Conference*, Fukui, Japan, Vol. 1, pp. 1136-1139.
9. Danny B. L. (1998), "Mobile Objects and Mobile Agents: The Future of Distributed Computing?", *Lecture Notes in Computer Science*, Vol. 1445, pp. 1-12.
10. Gary T., Rassul A., YuSong Z., and Farshad M. (2000), "Grid-Based Data Management in Distributed Simulation", *Proc. of the 33rd Annual Simulation Symposium*, Washington, pp. 1-7.
11. Ryan, P. and J. M. (1997), "Network Traffic Prediction for Distributed Interactive Simulation Exercises", *Proc. of SimTect 97*, Canberra, Australia
12. Bell, John L.(1999), *The Art of the Intelligible: An Elementary Survey of Mathematics in its Conceptual Development*, Kluwer.
13. McClave, J., Benson, P., and Sincich, T. (2004), *Statistics for Business and Economics*, 9th edn. Prentice-Hall.
14. Zeigler, B. P. (2003), "DEVS today: recent advances in discrete event-based information technology", *Proc. of the 11th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer Telecommunications Systems*, Arizona, pp. 148-161.
15. Navidi, W. and Camp, T. (2004), "Stationary distributions for the random waypoint mobility model", *IEEE Transactions on Mobile Computing*, Vol. 3, No. 1, pp. 99-108.
16. Spyros, G. M., Steven, C. W., and Rob J. H. (1997) *Forecasting: Methods and Applications*, 3rd ed. Wiley.



장 성 호 (ho7809@hanmail.net)

2004 용인대학교 컴퓨터 정보공학과 학사
 2006 인하대학교 컴퓨터 정보공학과 석사
 2006. 3~현재 인하대학교 정보공학과 박사과정

관심분야 : 분산 시뮬레이션, 병렬 컴퓨팅, 네트워크 관리, 소프트웨어 모델링



김 태 영 (silverwild@gmail.com)

2007 인하대학교 컴퓨터공학부 학사
 2009 인하대학교 정보공학과 석사
 2009~현재 인하대학교 정보공학과 박사과정

관심분야 : 시스템 모델링 & 시뮬레이션, 분산처리



이 종 식 (jslee@inha.ac.kr)

1993 인하대학교 전자공학과 학사
 1995 인하대학교 전자공학과 석사
 2001 미국 애리조나대 전기·컴퓨터공학과 박사
 2001~2002 캘리포니아 주립대학교 전기·컴퓨터공학과 전임강사
 2002~2003 클리블랜드 주립대학교 전기·컴퓨터공학과 조교수
 2003~2006 인하대학교 컴퓨터공학부 조교수
 2006~현재 인하대학교 컴퓨터공학부 부교수

관심분야 : 소프트웨어공학, 분산 컴퓨팅, 시스템 모델링 & 시뮬레이션