

이동 통신 시스템에서 사용자 이동성의 학습과 예측에 의한 이동성 데이터베이스의 실패 회복

(Failure Restoration of Mobility Databases by Learning and Prediction of User Mobility in Mobile Communication Systems)

길 준 민 † 황 종 선 ** 정 영 식 ***

(Joon-Min Gil) (Chong-Sun Hwang) (Young-Sik Jeong)

요 약 본 논문에서는 이동 통신 시스템 내에 존재하는 이동성 데이터베이스의 실패 처리를 위한 이동성 학습과 예측에 기반한 회복 기법을 제안한다. 이동 통신 시스템에서 이동성 데이터베이스는 사용자들에게 빠른 연결을 제공하기 위해 사용자의 현재 위치 정보를 유지해야 한다. 그러나, 이동성 데이터베이스의 실패는 사용자의 위치 정보를 잃어버리게 만든다. 결과적으로, 명백한 회복 과정 없이는 실패 상황에서 사용자의 호 요청은 거절된다. 따라서, 이동성 데이터베이스에 실패가 발생하였을 때, 실패에 효과적으로 대처할 수 있는 명백한 회복 기법이 사용자들에게 연속적인 서비스 가용성을 보장해 주기 위해서 필요하다. 본 논문의 회복 기법에서 사용되는 이동성 학습과 예측은 이동성 데이터베이스의 실패 후 시스템에 의해서 사용자의 위치를 파악하기 위한 기능을 담당한다. 실패 없는 연산 동안 사용자의 이동 패턴은 뉴로-퍼지 추론 시스템에 의해서 학습되며, 학습된 위치 정보는 실패 후 잃어버린 사용자의 위치를 파악하기 위해서 사용된다. 따라서, 본 논문의 회복 기법은 백업 과정과 검사점 정보를 저장하기 위해 필요한 추가적인 저장 공간을 요구하지 않기 때문에 검사점을 사용하는 이전의 접근방법과 다르다. 게다가, 성능 분석을 위한 시뮬레이션은 본 논문의 회복 기법이 실패 후 잃어버린 사용자의 위치 정보를 회복하는데 소요되는 비용을 검사점에 기반한 회복 기법과 비교하여 상당히 줄일 수 있음을 보여준다.

키워드 : 이동 통신 시스템, 이동성 데이터베이스, 실패 회복, 이동성 학습과 예측

Abstract This paper proposes a restoration scheme based on mobility learning and prediction in the presence of the failure of mobility databases in mobile communication systems. In mobile communication systems, mobility databases must maintain the current location information of users to provide a fast connection for them. However, the failure of mobility databases may cause some location information to be lost. As a result, without an explicit restoration procedure, incoming calls to users may be rejected. Therefore, an explicit restoration scheme against the failure of mobility databases is needed to guarantee continuous service availability to users. Introducing mobility learning and prediction into the restoration process allows systems to locate users after a failure of mobility databases. In failure-free operations, the movement patterns of users are learned by a Neuro-Fuzzy Inference System (NFIS). After a failure, an inference process of the NFIS is initiated and the users' future location is predicted. This is used to locate lost users after a failure. This proposal differs from previous approaches using checkpoint because it does not need a backup process nor additional storage space to store checkpoint information. In addition, simulations show that our proposal can reduce the cost needed to restore the location records of lost users after a failure when compared to the checkpointing scheme

Key words : Mobile Communication Systems, Mobility Databases, Failure Restoration, Mobility Learning and Prediction

† 비 회 원 : 고려대학교 기초과학연구소 연구원
jmgil@disys.korea.ac.kr

** 중신회원 : 고려대학교 컴퓨터학과 교수
hwang@disys.korea.ac.kr

*** 중신회원 : 원광대학교 컴퓨터 및 정보통신공학부 교수
ysjeong@wonms.wonkwang.ac.kr

논문접수 : 2001년 10월 4일
심사완료 : 2002년 3월 29일

1. 서론

이동 통신 환경(mobile communication environment)은 이동 사용자들이 시간과 장소에 제약 없이 통신이 가능한 환경에 기반을 두고 있기 때문에, 이동 사용자는 데이터를 주고받기 위하여 고정된 장소에 머무를 필요가 없다[1, 2]. 이러한 이동성을 지원하기 위해서는 사용자의 위치 정보는 호(call)가 설정되기 이전에 시스템에 알려져야 한다. 이동 통신 시스템에서 위치 관리(location management)는 사용자의 위치 정보를 관리하는 역할을 제공하며, 이동성 데이터베이스(mobility database)에서 이러한 위치 정보를 유지, 관리한다. 임의의 사용자를 대한 호 설정 요청이 있을 때, 이동성 데이터베이스에 저장된 위치 정보를 이용하여 호 설정이 이루어진다.

일반적으로 이동성 데이터베이스는 계층적으로 구성되어 있다. 사용자가 등록 영역(registration area)의 경계를 지나갈 때마다, 사용자의 새로운 위치 정보를 이동성 데이터베이스에 알려주기 위하여 위치 등록(location registration) 과정이 사용자에게 의해서 수행된다. 사용자가 피호출 되었을 때, 이동성 데이터베이스는 사용자가 현재 방문하고 있는 위치 정보를 얻기 위하여 질의를 받는다. 이러한 절차를 위치 질의(location query) 과정이라 하며 시스템에 의해서 수행된다.

위치 관리의 성능은 근본적으로 이동성 데이터베이스의 구조와 위치 관리 전략에 의해서 좌우되는데, 현재의 표준인 IS-41과 GSM의 네트워크 구조 하에서 위치 관리 비용을 최소화하기 위한 많은 연구가 있어 왔다[3, 4, 5, 6]. 이들 대다수의 연구는 주로 위치 등록과 위치 질의 비용을 최적화하는 방안에 초점을 맞추어 왔다. 그러나, 이동 통신 시스템에서 사용자의 위치 정보를 관리하는 이동성 데이터베이스에 대한 신뢰성(reliability)에 관련된 문제는 이동 환경에서 빈번히 발생하는 문제임에도 불구하고 이에 관련된 연구는 아직 미미한 수준이다. 이동성 데이터베이스에서 발생하는 잘못된 연산 및 비정상적인 동작 등으로 인하여 실패가 발생한다면, 이동성 데이터베이스에 저장된 위치 정보는 잃어버리게 될 것이고, 잃어버린 위치 정보에 해당되는 사용자에게 들어오는 호는 위치 정보의 부재로 인하여 거절된다. 그러면, 이동 통신 시스템은 더 이상 이들 사용자에게 들어오는 호에 대해서 어떠한 연결도 설정할 수 없다. 또한, 호 설정이 이루어진다 하더라도, 사용자는 많은 서비스 지연을 감수하여야 한다. 이는 사용자에게 대한 서비스 품질(quality of service)과 접근 가용성(access availability)을 급격하게 감소시켜 전체 시스템의 성능을 심각하게 저하시키는 요

인이 된다. 따라서, 이동성 데이터베이스의 실패 후에도 사용자들에게 연속적인 서비스 가용성을 보장해 주기 위하여 실패된 이동성 데이터베이스를 효과적으로 복구하는 명백한 회복 기법이 필요하다.

IS-41 표준[7]에서는 명백한 회복 기법 없이 수동적으로 사용자에게 의한 세 가지 사건(호 개시, 셀 교차, 위치 등록)에 의존하여 실패된 이동성 데이터베이스를 회복시킨다. 실패 후, 세 가지 사건을 통해서 노출된 위치 정보를 이용하여 실패된 이동성 데이터베이스는 재구성될 수 있지만, IS-41에서 이러한 회복 방법은 사용자의 사건 발생 빈도에 크게 의존하는 방법이다. 오랜 기간 동안 사용자의 사건이 발생되지 않는다면, 그 기간 동안에 사용자는 전혀 서비스를 제공받지 못한다. 한편, 또 다른 표준인 GSM[8]에서는 검사점 기법(checkpointing scheme)에 이용하여 실패 없는 연산 동안 주기적으로 이동성 데이터베이스를 안전한 저장소로 백업하고, 실패 발생 시 백업된 정보로 이동성 데이터베이스를 즉시 재구성하여 회복시킨다. GSM은 실패 후 명백한 회복에 의해서 서비스 가용성을 증대시킬 수 있지만, 회복을 위한 과정이 수동적으로 이루어져, 실패 후 위치 정보를 재구성하기 위한 비용이 많이 소요되는 단점이 있다. 특히, 사용자의 이동성이 큰 경우에는 회복되는 정보 중 쓸모 없는 정보의 발생 가능성이 대단히 높아짐으로써 올바른 위치 정보의 파악을 위하여 추가적인 페이징을 수행해야 한다. 따라서, IS-41과 GSM 표준에서의 회복 방법을 고려해 볼 때, 회복에 소요되는 비용의 최소화를 위해서는 사용자의 일련의 사건을 통한 회복 및 임의의 저장 장치로의 수동적 백업보다는 시스템에 의한 적극적인 회복 기법이 요구된다.

이동성 데이터베이스의 실패 후 재구성되는 위치 정보는 쓸모 없는 정보의 검증에 위한 사용자의 올바른 위치 파악을 목적으로 페이징이 수행되는데, 이동 통신 환경의 특성상 사용자는 그들의 이동성에 따라서 회복되는 위치 정보에 대응되는 위치에 상주하거나 그렇지 않을 수도 있다. 이동성이 큰 사용자일수록 회복되는 위치 정보에 상주하지 않을 가능성이 높아진다. 이러한 경우, 회복되는 위치 정보에 한번의 페이징 후 순차적으로 여러 번의 페이징 또는 극단적으로는 등록 영역 전체의 페이징을 통하여 사용자의 위치가 파악되어야 한다. 따라서, 실패 후 회복되는 위치 정보의 쓸모 없음을 검증하기 위해 사용되는 페이징 비용을 최소화하기 위해서는 사용자의 이동성이 실패 없는 연산의 수행동안 고려되어야 하며, 실패 후 이러한 정보를 이용하여 회복 과정에서 소요되는 추가적인 페이징 비용을 최소로 감소시

키는 것이 바람직하다.

본 논문에서는 이동성 데이터베이스의 실패 후 서비스 품질과 접근 가능성의 급격한 감소를 해결하고자 사용자 이동성을 고려한 회복 기법을 제안한다. 뉴로-퍼지 추론 시스템을 이용하여 사용자의 이동 패턴을 학습하고, 사용자의 이동 환경에 상관없이 사용자의 미래 위치의 예측을 통하여 실패된 이동성 데이터베이스를 회복시키는 접근 방법을 고려했다. 즉, 본 논문에서 제안하는 회복 기법은 사용자 이동성의 학습과 예측에 기반을 둔다. 실패 없는 연산 기간 동안 사용자가 새로운 위치 영역으로 진입하였을 때, 사용자의 움직임 패턴이 뉴로-퍼지 추론 시스템에 의해서 학습된다. 이동성 데이터베이스에 실패가 발생하였을 때, 실패 후 사용자가 상주할 만한 위치를 뉴로-퍼지 추론 시스템이 예측한 정보로 실패된 이동성 데이터베이스를 회복하도록 한다. 검사점을 사용하는 회복 기법과 달리, 본 논문에서 제안하는 회복 기법은 실패 없는 연산 기간 동안 수행되는 주기적 백업 등의 처리 과정이 필요하지 않아 시스템에 훨씬 적은 부담을 주며, 검사점을 저장하기 위한 추가적인 저장 공간을 요구하지 않는다. 또한, 실패 후 사용자의 정확한 위치 파악을 위해서 수행되는 페이지의 횟수를 줄일 수 있으므로 기존의 검사점 기법에 의한 회복 기법보다 회복에 소요되는 비용을 훨씬 줄일 수 있다. 아울러, 본 논문에서 제안하는 회복 기법의 성능은 검사점에 의한 회복 기법의 복구율과 달리 실패기간이 비교적 오래 지속됨에도 불구하고 사용자의 이동성에 상관없이 일정하게 높은 복구율을 보여 준다.

본 논문의 2장에서는 이동 통신 시스템에서의 위치 관리와 이동성 데이터베이스의 실패를 간략히 설명한다. 3장에서 실패된 이동성 데이터베이스의 회복을 위한 기존 연구들을 살펴보고 기존 기법들의 문제점을 기술한다. 4장에서는 본 논문에서 제안하는 회복 기법에 근간을 이루는 이동성 학습과 예측 시스템에 대해서 기술하고, 본 논문에서 제안하는 회복 기법은 5장에서 보여준다. 6장에서는 제안하는 회복 기법의 성능 평가를 위한 시뮬레이션을 기술하며 회복 과정에서 필요로 하는 페이지 비용 관점에서 검사점에 의한 회복 기법과 비교한다. 마지막으로, 8장에서 본 논문의 결론을 맺는다.

2. 이동 통신 시스템에서의 위치 관리와 이동성 데이터베이스의 실패

2.1 시스템 구조

본 논문에서 기반으로 하는 이동 통신 환경의 지리적

모델은 많은 육각형 셀(cell)들이 거대한 지역을 포함하는 구조이다. 대다수의 이동 통신 시스템들이 대역폭의 효율성을 위하여 이러한 셀룰러 구조를 사용하여 왔다 [1, 2]. 셀룰러 구조에서, 하나의 서비스 지역을 담당하는 하나의 셀은 무선 링크를 통하여 셀 내의 이동 사용자와 통신할 수 있는 기지국(Base Station; BS)을 가지고 있다. 몇몇 셀들은 등록 영역(Registration Area; RA)으로 묶여져 구성되어 있으며, 유선 네트워크와 인터페이스의 역할을 하는 MSC(Mobile Switching Center)와 연결되어 있다. MSC는 일반적으로 스위칭 기능을 가지고 있어서 위치 등록과 호 전달(call delivery)을 조율하는 역할을 담당하며, 시스템 내의 이동성 데이터베이스에 대한 접근을 갖는다. 현존하는 이동 통신 표준인 북미 표준의 IS-41과 유럽 표준의 GSM은 이동성 데이터베이스의 2 단계 계층 구조(two-tier hierarchical structure)에 기반을 두고 있다. 이들 두 표준은 사용자의 위치 정보를 관리하기 위한 이동성 데이터베이스로서 HLR(Home Location Register)과 VLR(Visitor Location Register)을 사용한다. HLR은 사용자의 ID와 디렉토리 정보, 프로파일 정보, 현재 위치 정보, 인증 정보 등과 같은 정보를 저장, 유지, 관리하며 이동 통신 시스템 내에 등록된 모든 사용자들에 대한 정보를 갖고 있는 전역적 데이터베이스이다. 반면, VLR은 사용자가 HLR이 담당하고 있는 RA 이외의 다른 RA를 방문하였을 때, 사용자가 현재 방문하고 있는 RA에 대한 위치 정보의 관리를 담당하는 지역적 데이터베이스이다. 그림 1은 본 논문에서 사용하는 2 단계 이동성 데이터베이스 구조에 기반을 둔 이동 통신 시스템의 구조를 보여준다.

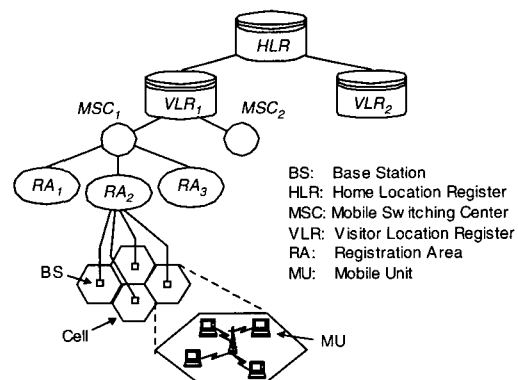


그림 1 이동 통신 시스템의 구조

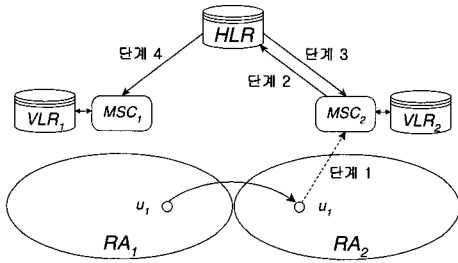


그림 2 위치 등록 과정

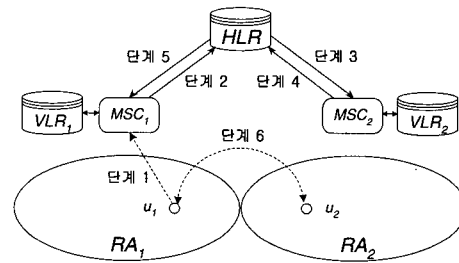


그림 3 위치 질의 과정

2.2 위치 등록과 위치 질의

위치 등록은 사용자가 새로운 RA로 이동하였을 때, 이동성 데이터베이스의 위치 정보를 갱신하는 과정이다. 아울러, 위치 질의는 임의의 사용자에게 들어오는 호를 설정해 주기 위한 과정을 수행한다.

위치 등록 과정은 그림 2로 설명되며 다음과 같은 단계들을 수행한다. 사용자 u_1 이 MSC_1 의 등록 영역 RA_1 에서 MSC_2 의 RA_2 로 이동하였을 때, u_1 은 MSC_2 로 위치 등록 메시지를 보낸다(단계 1). MSC_2 는 수신한 등록 메시지를 HLR 로 전달하고(단계 2), HLR 은 MSC_2 로 응답 메시지를 보낸다(단계 3). 등록 메시지를 받은 HLR 은 u_1 의 바뀐 위치 정보를 MSC_1 에 알린다. MSC_1 은 이 등록 메시지를 받자마자 VLR_1 에 있는 u_1 의 레코드를 삭제한다(단계 4). 또한, MSC_2 는 VLR_2 에 있는 u_1 의 레코드를 갱신한다.

그림 3은 위치 질의 과정을 나타내며 다음과 같은 단계들을 수행한다. 사용자 u_1 이 다른 사용자 u_2 를 호출하였을 때, u_1 은 MSC_1 으로 위치 질의 메시지를 보낸다(단계 1). 이때, MSC_1 은 사용자 u_2 의 위치 정보를 찾기 위하여 VLR_1 에 질의한다. u_2 의 위치 정보를 찾았다면, MSC_1 은 u_1 과 u_2 사이에 연결을 설정한다. 이 경우는 두 사용자 u_1 과 u_2 가 RA_1 에 상주함을 의미한다. u_2 의 위치 정보가 VLR_1 에 없다면, MSC_1 은 u_2 의 위치 정보를 찾기 위해서 HLR 에 질의한다(단계 2). HLR 은 u_2 가 현재 상주하고 있는 등록 영역을 질의에 의해서 얻어내고, MSC_2 로 라우트 요청 메시지를 보낸다(단계 3). 이때, MSC_2 는 u_2 의 TLDB (Temporary Location Directory Number)를 결정한다(단계 4). HLR 은 u_2 의 TLDB를 MSC_1 으로 전달한다(단계 5). 마지막으로 두 사용자 u_1 과 u_2 사이에 연결이 설정된다(단계 6)

2.3 이동성 데이터베이스의 실패

2.1절과 2.2절에서 설명된 위치 등록과 위치 질의 과정을 수행하는데 있어서 여러 이동성 데이터베이스들

중 어느 하나에 실패가 발생한다면, 사용자의 정확한 위치 정보는 파악될 수 없을 뿐만 아니라, 최신의 위치 정보 또한 갱신될 수 없다. 결론적으로, 실패 후 임의의 사용자에게 들어오는 호에 대한 연결 설정은 이루어지지 않는다.

일반적으로 명백한 회복 기법 없이 사용자의 위치 정보는 다음 세 가지 사건 중에 하나에 의해서 회복될 수 있다.

- 자동적인 위치 등록(autonomous location registration): 사용자는 주기적으로 시스템에게 그들의 현재 위치를 알린다.
- 호 개시(call origination): 시스템은 사용자의 현재 위치를 호 개시 때 파악할 수 있다.
- 등록 영역 교차(registration boundary crossing): 사용자가 등록 영역을 넘어설 때 반드시 시스템에게 등록 메시지를 보내야만 한다.

위의 세 가지 사건은 이동성 데이터베이스의 실패 후 명백한 회복 기법 없이 실패된 이동성 데이터베이스를 회복하기 위해서 사용된다. 그러나, 회복에 소요되는 지연과 비용은 사용자에 의한 위의 세 가지 사건의 발생 빈도에 크게 의존한다. 극단적으로 실패 후 위의 세 가지 사건 중 어떠한 사건도 발생하지 않는다면, 실패된 이동성 데이터베이스는 영원히 회복될 수 없다. 또한, 위의 세 가지 사건 중 어떤 사건이 발생하더라도 오랜 시간이 지난 후에 발생한다면, 이 시간 동안에는 어떠한 서비스도 사용자에게 제공되지 않는다.

따라서, 위에서 기술된 수동적 회복 기법에 의존하기 보다는 실패 후 잃어버린 위치 정보를 재구성하기 위해서 명백한 회복 기법이 요구된다. 결론적으로 사용자에 의한 일련의 사건을 통한 이동성 데이터베이스의 회복 보다는 시스템에 의해서 수행되는 회복을 통하여 빠른 시간에 사용자의 위치 정보를 능동적으로 재구성해야 하며, 이로 인하여 사용자는 서비스 가용성을 보장 받을 수 있다.

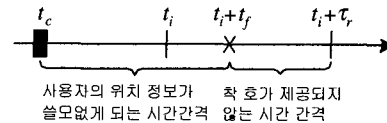
3. 기존 연구 및 문제점

IS-41과 GSM 표준에서의 이동성 데이터베이스의 구조는 HLR과 VLR로 구성되는 2 단계 데이터베이스 구조에 기반을 하고 있다. IS-41 표준은 시스템 내에 명백한 회복 기법을 제공하지 않으며 앞서 기술한 사용자에 의한 세 가지 사건을 통해서 실패된 이동성 데이터베이스를 회복한다. IS-41 표준과 달리, GSM은 회복을 위한 능동적 메커니즘[8]이 존재한다. GSM에서는 실패된 이동성 데이터베이스의 회복을 위하여 다음과 같은 과정을 수행한다: 실패 없는(failure-free) 연산 동안, 이동성 데이터베이스 내의 위치 정보들이 이동성 데이터베이스(HLR과 VLR) 근처에 위치하고 있는 안전한 저장소로 주기적 검사점이 취해진다. 이동성 데이터베이스에 실패가 발생한 후, 이동성 데이터베이스 내의 위치 정보를 안전한 저장소에 저장된 백업 정보로 즉각적으로 재구성함으로써, 실패된 이동성 데이터베이스는 회복된다. 그러나, 백업된 위치 정보들이 모두 정확하지는 않다. 이를 좀더 구체적으로 설명하면, 검사점이 취해진 시간과 실패 시간 사이에 갱신된 위치 정보는 백업에 반영되지 않기 때문에, 백업된 위치 정보가 쓸모 없게(obsolete)될 가능성을 지니고 있다. 이러한 상황에서, 사용자에 대한 호가 발생하였을 경우, 백업된 위치 정보와 사용자가 현재 상주하고 있는 위치가 서로 일치하지 않아 정확한 위치 정보를 시스템이 파악할 수 없다. 결과적으로, 연결 설정은 이루어지지 않고, 사용자에 대한 호는 잃어버리게 된다. 쓸모 없는 위치 정보는 앞서 기술한 사용자에 의해서 발생하는 자동적인 위치 등록, 호 시작 및 등록 영역 교차를 통하여 갱신될 수 있더라도, 이러한 세 가지 사건조차도 오랜 기간 동안 발생하지 않는다면, 잃어버리게 되는 호는 그 기간에 비례하여 점점 늘어나게 된다. 이것은 서비스 가용성을 심각하게 저하시킨다.

이러한 문제점을 해결하기 위해서 몇몇 회복 기법들이 [9, 10]에서 제안되어 왔는데, 이들 연구들은 주로 페이징 비용과의 균형을 맞추기 위해서 최적 검사점 간격(optimal checkpoint interval)의 유도에 초점을 맞추었다. 그러나, 이들 방법은 사용자의 다양한 이동성 때문에 모든 이동 사용자에게 동일한 검사점 간격을 적용하기에는 무리가 따른다. 게다가, 사용자의 이동성은 실패 후 잃어버린 사용자의 위치를 찾는 데 소요되는 비용과 지연에 밀접하게 관련되어 있다. 예를 들어, 빠르게 움직이는 사용자는 마지막 검사점이 취해진 위치에 상주할 가능성은 매우 희박하다. 이 경우, 시스템은 마치

막으로 검사점이 취해진 셀에 대한 잘못된 페이징 후에 등록 영역 내의 전체 셀을 페이징하여 사용자의 위치를 파악하는데, 이것은 페이징 비용의 증가를 초래한다. 따라서, 검사점 기법은 회복 과정의 효율성 증대를 위한 적절한 접근 방법이라 할 수 없다.

이에 대한 구체적인 설명을 위해 그림 4로 표현되는 시간 흐름을 살펴보자. 그림 4에서 시간 t_c 는 이동성 데이터베이스에 대해서 검사점이 취해진 시간을 나타낸다. 검사점이 취해진 후, 사용자는 검사점이 취해진 등록 영역을 벗어나 시간 t_i 에 새로운 등록 영역으로 움직인다. t_r 은 새로운 등록 영역에서 사용자의 상주시간을 나타낸다. 즉, 그림 4의 시간 흐름에서 사용자는 시간 t_i 에 새로운 등록 영역으로 진입하고 시간 $t_i + \tau_r$ 에 그 등록 영역을 벗어난다. 여기서, 이동성 데이터베이스의 실패는 시간 t_i 와 $t_i + \tau_r$ 사이의 임의의 시간 $t_i + t_f$ 에서 발생한다고 가정한다($0 \leq t_f \leq \tau_r$).



- t_c : 검사점이 취해진 시간
- t_i : 마지막 위치 갱신이 발생한 시간
- $t_i + t_f$: 이동성 데이터베이스에 실패가 발생한 시간
- $t_i + \tau_r$: 사용자가 자신의 위치를 등록한 시간

그림 4 시간 흐름

이동성 데이터베이스의 실패 후, 시간 $t_i + t_f$ 와 $t_i + \tau_r$ 의 사이에는 어떠한 호도 해당 사용자로 전달될 수 없는데, 이는 사용자의 위치 등록이 수행되어야 사용자의 위치가 시스템으로 알려질 수 있기 때문이다. τ_r 이 비교적 크다면, 즉 사용자가 오랜 시간이 지난 후에 자신의 위치를 등록한다면, 비 서비스 기간은 점점 늘어난다. 시간 $t_i + t_f$ 에서 실패가 발생했을 때, 이동성 데이터베이스는 안전한 저장소로부터 즉각적으로 회복된다. 이 시점에서, 시스템은 회복된 위치 정보의 쓸모 없음을 검증하기 위해서 회복된 위치에 대해서 페이징을 시작한다. 만일 t_c 와 $t_i + t_f$ 의 간격이 매우 길다면, 검사점을 안전한 저장소로 백업하는 횟수가 줄어들기 때문에 실패 없는 연산 동안의 지연은 줄어들 수 있다. 그러나, 대부분의 사용자는 시간 t_c 에서 검사점이 취해진 위치를 벗어나 새로운 위치로 이동하게 될 가능성이 매

우 높아진다. 이 경우, 사용자는 검사점이 취해진 위치에 대응되는 셀에 상주하지 않으므로, 검사점이 취해진 위치에 대한 페이징만으로는 사용자의 정확한 위치를 파악하지 못하게 된다. 이때, 시스템은 첫번째 잘못된 페이징 이후에 해당 등록 영역 내의 전체 셀에 대한 페이징을 수행해야만 한다. 따라서, 사용자의 이동성의 커짐에 따라 이러한 불필요한 페이징은 필연적으로 점점 많아지게 되고 실패 후 사용자의 정확한 위치를 찾기 위한 비용을 증가시키는 요인이 된다. 반면, 좀 더 정확한 위치 정보를 파악하기 위해 시간 t_c 와 $t_i + t_f$ 사이의 간격을 줄임으로서 검사점 간격을 조밀하게 할 수 있다. 그러나, 안전한 저장소로 백업하는 횟수가 늘어나기 때문에, 실패 없는 연산 동안의 지연이 늘어난다.

이상에서 살펴본바와 같이, 이동 통신 시스템에서 이동성 데이터베이스의 실패 후 명백한 회복 기법이 요구되며, 사용자의 일련의 사건에 의존하여 실패된 이동성 데이터베이스를 회복하는 것보다는 시스템 내에서 회복을 위한 처리 과정이 능동적으로 수행되는 것이 바람직하다. 아울러, 사용자의 이동성은 회복에 소요되는 비용과 밀접한 관계가 있으므로, 실패 없는 연산 기간 동안에 사용자의 이동성이 고려되는 것이 중요하다. 실패 후, 회복되는 위치 정보가 사용자가 상주할 만한 위치와 일치되도록 하여 불필요한 페이징 횟수를 줄임으로서 회복 과정의 효율성을 높여야 할 것이다. 이를 위해 본 논문에서는 사용자 이동성의 학습을 실패 없는 연산에 도입하고 실패 후 이동성 예측에 의해서 회복 과정의 효율성을 높이고자 한다. 다음 장에서는 제안하는 이동성 데이터베이스의 회복 기법에서 사용되는 사용자 이동성의 학습과 예측을 기술한다.

4. 사용자 이동성의 학습과 예측

사용자 이동성 학습과 예측은 사용자의 이동성이 반영된 이동 궤적을 이동 속도와 방향으로 표현하고 현재와 과거의 이동 속도와 방향을 기반으로 미래 위치를 예측한다. 본 논문에서는 이동성 학습과 예측을 위한 모델링 방법으로서 뉴로-퍼지 추론 시스템을 이용하는데, 사용자의 이동 위치의 예측을 위한 기존 기법들[11, 12, 13]이 단순 휴리스틱에 기반을 하고 있는 점에서 동기가 부여되었다. 이들 기법들은 현재 이동 상황에 대한 단순한 관찰에 의존하기 때문에, 사용자의 현재 위치가 시스템에 미리 알려지지 않은 경우 좋지 못한 예측 결과를 초래한다. 그러나, 저자가 [14, 15]에서 제안한 이동성 학습과 예측 시스템에서는 뉴로-퍼지 추론 시스템에 의해서 사전에 알려지지 않은 이동 위치라 할지라도,

미래 위치에 대한 적절한 예측을 수행한다. 이는 특히 다양하게 변하는 이동 상황에 적응적으로 대처할 수 있다. 이동성 학습과 예측 시스템을 설명하기 앞서 이 시스템에서 기본적으로 이용되는 뉴로-퍼지 추론 시스템 대해서 기술한다.

4.1 뉴로-퍼지 추론 시스템

본 논문에서의 뉴로-퍼지 추론 시스템(neuro-fuzzy inference system; 이하 NFIS)은 퍼지 추론 시스템에 신경망의 학습 능력(learning capability) 및 환경 변화에 대한 적응성(adaptability)을 융합한 형태를 갖는다. 본 논문에서 사용하는 퍼지 추론 시스템은 Sugeno의 퍼지 시스템에서 퍼지 규칙의 결론부를 상수 값으로 제한함으로써 비퍼지화 과정의 계산량을 줄인 단순 퍼지 추론 시스템(simplified fuzzy inference system)이다[16]. 본 논문에서의 단순 퍼지 추론 시스템은 다음과 같은 형태의 퍼지 규칙을 갖는다:

Rule i : If x_1 is A_1^i and, ..., and x_d is A_d^i then y is w_i (1)

여기서, A_j^i 는 $i(=1, 2, \dots, n)$ 번째 퍼지 규칙의 전체부에 있는 입력 변수 $x_j(j=1, 2, \dots, d)$ 에 대한 퍼지 집합을 나타내며, w_i 는 i 번째 퍼지 규칙에서 결론부의 실수값을 나타낸다. (1)과 같은 형태의 퍼지 규칙을 갖는 퍼지 추론 시스템은 실수 벡터 입력 $\bar{x} = [x_1, x_2, \dots, x_d]$ 에서 실수 출력 y 로 사상하는 함수 $f: \bar{x} \rightarrow y$ 로 표현하면 다음과 같다:

$$f(\bar{x}) = \frac{\sum_{i=1}^n \mu_i \cdot w_i}{\sum_{i=1}^n \mu_i}, \quad \mu_i = \prod_{j=1}^d \mu_{A_j^i}(x_j) \quad (2)$$

여기서, $\mu_{A_j^i}$ 는 퍼지 집합 A_j^i 에 대한 퍼지 소속 함수를 나타내며 μ_i 는 i 번째 퍼지 규칙의 퍼지 소속 값을 나타낸다. (2)의 퍼지 추론 시스템은 입력 실수값의 퍼지화로서 단일값 퍼지화(singleton fuzzifier), 퍼지 함축 연산자로서 product-inference 연산자, 비퍼지화로서 weighted-average 방법을 사용하여 퍼지 추론 시스템의 연산 과정을 단순화한다. 퍼지 소속 함수는 응용 문제에 따라 삼각형(triangular), 사다리꼴(trapezoid), 가우시안(Gaussian) 등 여러 가지 형태 중에 하나를 사용할 수 있다. 본 논문에서는 가우시안 함수를 퍼지 소속 함수로 사용한다. 가우시안 함수는 연속이고 미분 가능한 성질을 갖고 있어서 학습 규칙에 적용될 수 있다. 본 논문에서 사용되는 가우시안 퍼지 소속 함수의 정의는 다음과 같다:

$$\mu_{A_j^i}(x_j) = \exp \left[-\frac{1}{2} \cdot \left(\frac{x_j - c_j^i}{\sigma_j^i} \right)^2 \right] \quad (3)$$

여기서, c_j^i 과 σ_j^i 은 각각 i 번째 퍼지 규칙의 전제부에서 가우시안 퍼지 소속 함수의 중심값과 퍼진 정도를 나타낸다.

(3)에서 퍼지 소속 함수의 파라미터(c_j^i , σ_j^i)와 퍼지 규칙 결론부의 실수값인 w_i 는 역전파 학습 알고리즘(back-propagation learning algorithm)을 이용하여 학습 오차(learning error)($e = 1/2 \sum_p (y_p - \hat{y}_p)^2$)가 최소가 되도록 조정한다[16]:

$$c_j^i(t+1) = c_j^i(t) - \eta \cdot \frac{\partial e}{\partial \mu_i} \cdot \frac{\partial \mu_i}{\partial A_j^i} \cdot \frac{\partial A_j^i}{\partial c_j^i} \\ = c_j^i(t) - \eta \cdot (y_p - \hat{y}_p) \cdot \left((w_i(t) - \hat{y}_p) / \left(\sum_k \mu_k \right) \right) \cdot \mu_i \cdot \left(\frac{x_j - c_j^i(t)}{(\sigma_j^i(t))^2} \right) \quad (4)$$

$$\sigma_j^i(t+1) = \sigma_j^i(t) - \eta \cdot \frac{\partial e}{\partial \mu_i} \cdot \frac{\partial \mu_i}{\partial A_j^i} \cdot \frac{\partial A_j^i}{\partial \sigma_j^i} \\ = \sigma_j^i(t) - \eta \cdot (y_p - \hat{y}_p) \cdot \left((w_i(t) - \hat{y}_p) / \left(\sum_k \mu_k \right) \right) \cdot \mu_i \cdot \left(\frac{(x_j - c_j^i(t))^2}{(\sigma_j^i(t))^3} \right) \quad (5)$$

$$w_i(t+1) = w_i(t) - \eta \cdot \frac{\partial e}{\partial w_i} \\ = w_i(t) - \eta \cdot \left(\mu_i / \sum_k \mu_k \right) (y_p - \hat{y}_p) \quad (6)$$

여기서, y_p 는 p 번째 입력력 패턴에 대한 실제 출력값, \hat{y}_p 는 p 번째 입력력 패턴에 대한 퍼지 시스템의 출력값, μ_i 는 i 번째 퍼지 규칙의 신뢰도(support value), 그리고 η 는 학습률(learning rate)이다.

4.2 이동성 학습과 예측 시스템

이동성 학습과 예측 시스템은 사용자의 이동성이 반영된 이동 경로를 이동 속도와 방향으로 표현하고 사용자의 미래 위치를 예측하는 시스템으로서 현재와 과거 위치의 학습 정보를 이용하여 미래 위치를 예측한다. 이 시스템은 시간 t 에서 사용자의 이동 상태를 다음과 같이 정의한다: $S^t = (v^t, \theta^t)$. 여기서, v^t 와 θ^t 는 각각 시간 t 에서의 이동 속도와 이동 방향을 나타낸다. 그리고, 사용자의 이동 상태는 고정된 시간 주기 τ 동안 변한다고 가정한다. 이러한 상태 정의에 의해서 이동성 학습 및 예측 시스템은 현재와 과거의 이동 상태로부터 미래 이동 상태로 사상하는 함수로 표현되며 다음과 같이 정의된다:

$$NFIS(S^t, S^{t-1}, \dots, S^{t-(h-1)}) \mapsto S^{t+1} \quad (7)$$

여기서, $NFIS$ 는 뉴로-퍼지 추론 시스템을 나타내며, h 는 현재와 과거의 이동 상태들의 개수를 나타내며 뉴로-퍼지 추론 시스템에서 입력 변수의 개수이다. (7)은 미래 이동 상태 S^{t+1} 을 예측하기 위해서 현재와 과거

의 이동 상태 $S^t, S^{t-1}, \dots, S^{t-(h-1)}$ 을 이용한다.

사용자의 수일, 수개월 동안의 이동 상태로부터 $NFIS$ 에서의 퍼지 규칙을 구성하기 위해서 본 논문에서는 사용자의 이동 상태의 유사 정도에 따라서 퍼지 규칙의 자동적인 생성을 가능하게 하는 클러스터링 방법(clustering method)[16]을 이용한다. 본 논문의 클러스터링 방법은 수일, 수개월 동안의 이동 상태들을 몇 개의 클러스터(cluster)로 나누고 각 클러스터에 속하는 이동 상태는 다른 클러스터에 속하는 이동 상태들보다 강한 유사성(similarity)을 갖도록 한다. 하나의 클러스터는 클러스터를 대표하는 중심을 갖는다. 클러스터의 중심은 퍼지 규칙의 전제부의 중심값으로 사용된다. M 개의 클러스터 중 i 번째 클러스터의 중심을 \bar{R}_i ($= [R_i^1, R_i^2, \dots, R_i^h]$)으로 하고, 시간 t 에서의 이동 상태 벡터를 $\bar{S}^t (= [S^t, S^{t-1}, \dots, S^{t-(h-1)}])$ 라 하였을 때, \bar{R}_i 와 \bar{S}^t 간의 유사성 측정 방법으로서 다음과 같은 거리 측정 방법을 이용한다:

$$D_i = |\bar{S}^t - \bar{R}_i| \quad (8)$$

(8)에서 D_i 가 작으면 작을수록 이동 상태 벡터와 클러스터간의 유사성이 강함을 나타낸다. 반면, D_i 가 크면 클수록 이동 상태 벡터와 클러스터간의 유사성이 약함을 보여준다. 그러면, 이동 상태 벡터와 클러스터간의 유사성을 판단하는 기준이 필요한데, 본 논문에서는 반경(radius) r 을 정의하여 D_i 가 r 보다 작다면, \bar{R}_i 와 \bar{S}^t 간의 유사성이 강하다고 정의하고, 그렇지 않다면 유사성이 약하다고 정의한다. 이와 같은 유사성 판단 기준에 기초하여, 사용자의 이동 상태는 \bar{R}_i 와 \bar{S}^t 간의 유사성과 반경 r 에 의해서 다음과 같이 세 가지 경우로 나누어진다:

- 이동 상태 벡터가 클러스터(퍼지 규칙)로 정의되지 않은 경우
- 이동 상태 벡터가 클러스터로 정의된 경우
- 이동 상태 벡터가 클러스터로 정의되었지만 불필요한 클러스터인 경우

첫번째 경우는 사용자의 이동 상태 벡터가 $NFIS$ 에 이미 정의된 클러스터와 대응되지 않고 별개의 이동 상태 벡터로서 $NFIS$ 에 새로운 클러스터로 추가된다. 이 경우는 $D_i > r$ 인 경우를 나타내며, 다음과 같이 새로운 클러스터에 대응되는 퍼지 규칙의 전제부와 결론부 실수값을 생성한다:

$$A_j^i = \exp \left[-\frac{1}{2} \cdot \left(\frac{x_j - S^{t-j}}{r} \right)^2 \right], \quad w_i = S^{t+1} \quad (9)$$

여기서, A_j^i 는 i 번째 퍼지 규칙의 j ($= 0, 1, \dots, h-1$) 번째 입력변수에 대한 퍼지 소속 함수를 나타낸다. 퍼지 소속 함수, A_j^i 의 중점값(c_j^i)과 퍼진 정도(σ_j^i)는 각각 S^{t-j} 와 r 로 대체되며 i 번째 퍼지 규칙의 결론부 실수값(w_i)는 S^{t+1} 로 대체된다.

두번째 경우는 사용자의 이동 상태 벡터가 이미 정의된 클러스터들 중 하나 이상의 클러스터와 유사성을 갖는 경우로서 유사성을 갖는 클러스터들에 대해서 가중치를 증가시켜서 동일한 시간과 위치에서 같은 이동 상태를 보일 때 **NFIS**가 다른 이동 상태 벡터보다도 더 나은 예측 결과를 보이도록 하는 역할을 한다. $D_i \leq r$ 가 이 경우를 나타낸다. 클러스터에 속하는 이동 상태 벡터들이 퍼지 규칙의 결론부 실수값에 영향을 미치도록 퍼지 규칙의 결론부 실수값을 다음과 같이 클러스터에 속하는 미래 이동 상태의 평균값을 취하도록 조정한다:

$$w_i = \frac{\sum_{k=1}^m S^{t+1}}{m_i} \quad (10)$$

여기서, w_i 는 i 번째 클러스터에 대응되는 퍼지 규칙의 결론부 실수값을 나타내고, m_i 는 i 번째 클러스터에 속하는 이동 상태 벡터의 개수이다.

세번째 경우는 사용자의 이동 상태 벡터가 퍼지 규칙(클러스터)으로 포함되었지만, 퍼지 규칙이 사용자의 최근 이동 상태를 반영하지 못하기 때문에 현재 **NFIS**에서 불필요한 퍼지 규칙으로 고려되는 경우를 나타낸다. 따라서, 이러한 퍼지 규칙은 적절한 이동 환경에 맞추어서 제거되어야 한다. 본 논문에서는 불필요한 퍼지 규칙을 제거하기 위해서 다음과 같은 에이징(aging) 방법을 이용한다:

$$L(age_i) = \exp[-\alpha \cdot age_i]$$

$$\text{Rule } i = \begin{cases} \text{young} & \text{if } L(age_i) \geq \beta \\ \text{old} & \text{if } L(age_i) < \beta \end{cases} \quad (11)$$

여기서, age_i 는 i 번째 퍼지 규칙의 나이를 나타내며, α 는 퍼지 규칙의 나이를 조정하는 파라미터이다. 그리고 β 는 퍼지 규칙의 제거를 판단하는 파라미터이다.

퍼지 규칙의 나이는 다음과 같이 이동 상황에 따라서 갱신된다: 현재 이동 상태에 대해서 가장 강한 유사성을 갖는 클러스터가 반경보다 작다면, 해당 클러스터에 대응되는 퍼지 규칙의 나이를 0으로 설정함으로써 클러스터를 유지시키도록 한다. 반면, 현재 이동 상태에 가장 강한 유사성을 갖는 클러스터가 반경보다 크다면, 해당 클러스터에 대응되는 퍼지 규칙의 나이를 1씩 증가시킨다. 나이가 많은 퍼지 규칙은 나이가 적은 퍼지 규칙보

다 상대적으로 최근 이동 상태에 대한 반영이 적다는 것을 의미한다. 따라서, 나이가 적은 퍼지 규칙은 최근 이동 상태를 반영하므로 퍼지 규칙으로서 유지하는 것이 바람직하다. 반면, 나이가 많은 퍼지 규칙은 최근 이동 상태를 반영하지 못하므로, 최근의 이동 상태에 대해서는 불필요한 퍼지 규칙으로서 잘못된 예측 결과를 초래할 수 있다. 이러한 퍼지 규칙의 제거를 판단하는 파라미터로서 β 를 도입한다. β 값을 기준으로 퍼지 규칙은 나이가 적은 퍼지 규칙과 나이가 많은 퍼지 규칙으로 나누어진다. 나이가 적은 퍼지 규칙은 최근 이동 상태를 반영하므로 퍼지 규칙으로서 사용되는 것이 바람직하다. 반면, 나이가 많은 퍼지 규칙은 β 의 값 기간동안 사용자의 이동 경로에 단지 한번 발견되는 이동 상태로부터 생성된 퍼지 규칙이다. 이는 퍼지 규칙에 최근 이동 상태가 올바르게 반영되지 못함을 의미하므로 퍼지 규칙 베이스로부터 삭제되어야 한다. 본 논문에서는 (11)에 의해서 나이가 많은 퍼지 규칙으로 판단되는 퍼지 규칙을 불필요한 퍼지 규칙으로 삭제한다.

이제까지 사용자의 이동 패턴을 모델링하고 예측하기 위해서 **NFIS**를 이용한 이동성 학습과 예측 시스템을 살펴보았다. 다음 장에서 이동성 학습과 예측에 의한 이동성 데이터베이스의 회복 기법을 제안한다.

5. 이동성 학습과 예측에 의한 이동성 데이터베이스의 회복

이 장에서는 이동 데이터베이스의 실패 후에도 사용자에게 서비스 가용성을 보장해 주는 이동성 데이터베이스의 회복 기법을 기술한다. 본 논문에서 제안하는 이동성 데이터베이스의 회복 기법은 4장에서 기술된 이동성 예측을 이동성 데이터베이스의 실패 후 회복 과정의 효율성을 위하여 사용한다. 제안하는 회복 기법은 실패

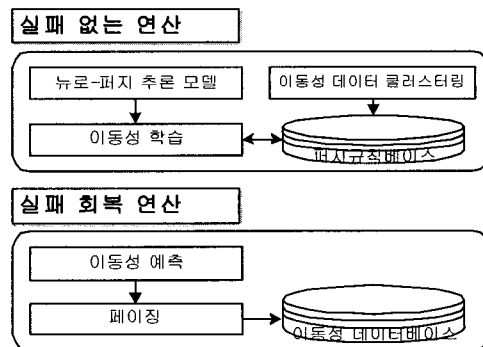


그림 5 제안하는 회복 기법의 구조

없는 연산(failure-free operation)과 실패 회복 연산(failure recovery operation)으로 구성되며, 그림 5는 본 논문에서 제안하는 회복 기법의 전반적인 구조를 보여준다. 다음에서 본 논문의 회복 기법을 각 연산별로 나누어서 기술한다.

5.1 실패 없는 연산

본 논문에서 제안하는 회복 기법은 단일 결함 모델(single failure model)로서 이동성 데이터베이스는 고장-멈춤(fail-stop)이라고 가정한다. 따라서, 이동성 데이터베이스의 실패 후 잃어버린 위치 정보들은 4장에서 기술한 이동성 학습과 예측 시스템에서 의해서 즉각적으로 회복된다.

실패 없는 연산 동안 제안하는 회복 기법은 이동성 학습에 의해서 사용자의 이동 패턴을 학습한다. 이동성 학습을 위한 입력 값으로 사용자의 현재 이동 방향과 속도가 고려된다. 기본적으로 제안하는 회복기법은 τ_r 시간 간격으로 사용자의 현재 위치를 이동성 데이터베이스에 알린다는 가정에 의해서 학습을 위한 정보를 추출한다. 무선 링크의 소모를 최소로 하기 위하여, 호 개시, 핸드오프, 위치 등록 등과 같은 메시지 교환이 일어날 때 얻어지는 사용자의 위치적 변화로부터 학습을 위한 정보를 또한 추출한다. MSC가 이러한 종류의 메시지 교환을 탐지하여 HLR에 위치하고 있는 이동성 학습 및 예측 시스템에게 사용자의 위치 정보를 알려준다. 이때, NFIS가 최신의 위치 정보를 이용하여 사용자의 이동 패턴을 학습한다.

실패 없는 연산 동안에 수행되는 이동성 학습 및 예측은 사용자의 이동 환경을 올바르게 반영하는 퍼지 규칙을 만들기 위한 사용자의 이동 패턴의 학습 과정과 사용자의 미래 위치를 예측하기 위한 예측 과정으로 구성되어 있으며, 다음과 같은 단계들을 수행한다.

시간 t 에서 사용자의 현재와 과거 이동 상태로 구성된 이동 상태 벡터를 $\vec{S}^t = [S^t, S^{t-1}, \dots, S^{t-(h-1)}]$ 이라고 가정하자. 시간 $t+1$ 에 대한 사용자의 이동 상태 S^{t+1} 을 NFIS에 의해서 예측하고 이를 기반으로 사용자의 이동 패턴을 학습하고자 한다.

[단계 1] 사용자의 첫번째 이동에 대한 이동 상태 벡터(\vec{S}^1)로부터 첫번째 클러스터 중심, $\vec{R}_1 = [S^1, S^0, S^{-(h-2)}]$ 을 만들고 $w_1 = S^0$ 라고 놓는다. 첫번째 클러스터의 나이를 $age_1 = 0$ 으로 놓는다. 또한, 클러스터의 반경 r 을 설정한다. 첫번째 이동 상태 벡터에는 사용자의 과거 이동 상태를 반영할 수 없으므로 무작위 값으로 설정한다. 이 과정은 h 번째 이동 상태까지 반복된다.

[단계 2] 사용자의 t 번째 이동에 대한 이동 상태 벡터, \vec{S}^t 을 고려해 보자. 여기서, M 개의 클러스터($\vec{R}_1, \vec{R}_2, \dots, \vec{R}_M$)가 설정되었고 i 번째 클러스터의 중심을 \vec{R}_i 라고 하자($i = 1, 2, \dots, M$). (8)에 의해서 \vec{S}^t 와 \vec{R}_i 간의 유사성을 모든 클러스터에 대해서 계산하고, 가장 강한 유사성을 갖는 클러스터의 중심을 \vec{R}_{max} 라고 하자.

① 사용자의 이동 상태 벡터가 클러스터(퍼지 규칙)으로 정의되지 않은 경우, 즉 $D_{max} > r$ 이라면, $\vec{R}_{M+1} = [S^t, S^{t-1}, \dots, S^{t-(h-1)}]$, $w_{M+1} = S^{t+1}$ 라고 놓고, $age_{M+1} = 0$ 으로 놓는다.

② 사용자의 이동 상태 벡터가 클러스터로 정의된 경우, 즉 $D_{max} \leq r$ 이라면, (9)에 의해서 퍼지 규칙의 결론부 실수값(w_{max})을 조정하고, $j = 1, 2, \dots, M$, $j \neq max$ 에 대해서 $age_j = age_j + 1$ 라고 놓는다.

[단계 3] NFIS에 존재하는 클러스터 중 불필요한 클러스터를 제거한다. 즉, i 번째 클러스터의 나이, age_i 에 대해서 $L(age_i) < \beta$ 이라면, \vec{R}_i 를 제거한다($i = 1, 2, \dots, M$).

[단계 4] (2)에 의해서 t 번째 이동 상태 벡터, \vec{S}^t 에 대해서 미래의 이동 상태 \vec{S}^{t+1} 을 NFIS에 의해서 예측한다.

[단계 5] t 번째 이동 상태에 대한 NFIS의 이동 예측 결과(\vec{S}^{t+1})와 실제 이동 결과(S^{t+1})를 (4)-(6)에 적용하여 학습 횟수만큼 NFIS의 파라미터(c_j^i , σ_j^i , w_i)를 조정한다.

[단계 6] 단계 2-5를 반복 수행한다.

5.2 이동성 데이터베이스의 실패 후 회복 과정

이동성 데이터베이스에 실패가 발생한 후, 실패된 이동성 데이터베이스는 실패 이전에 유지하고 있었던 위치 정보로 사용자에게 의한 수동적 회복보다는 시스템에 의한 능동적 회복 과정을 통하여 회복되어야 한다. 본 논문에서는 사용자의 잃어버린 위치 정보를 회복시키기 위한 방법으로 이동성 예측으로의 접근 방법을 모색하였으며, 실패 후 사용자가 상주할 위치를 NFIS에 의해서 예측된 정보를 이용하여 파악함으로써 실패 후 잃어버린 위치 정보를 회복한다. 이동성 데이터베이스의 실패 후 이동성 예측에 의한 회복 과정은 다음과 같다: 이동성 데이터베이스에 실패가 발생한 시점을 t_f 라 하고, 이 시점에서의 이동 상태 벡터를 \vec{S}^{t_f} 라 하였을 때, 시간 t_f 이후의 임의의 미래 시간 t_p 에 대한 이동 상태 \vec{S}^{t_p} 를 NFIS를 이용하여 예측해 보자($t_f < t_p$). 시간 t_f 와 t_p 사이의 임의의 미래 시간 t 에서는 미래 이동 상태 벡터 \vec{S}^t 를 구성하기 위해서 더 이상 과거 및 현재

이동 정보를 사용할 수 없다($t_f < t < t_p$). 따라서, 본 논문에서는 이동성 데이터베이스의 실패 시간 t_f 이후에 미래 시간 t_p 의 이동 상태를 예측하기 위하여 시간 t_p 까지 **NFIS**에 의해서 예측된 위치 정보를 이용하여 미래 이동 상태 벡터 \overline{S}^t 를 다음과 같이 구성한다:

$$\overline{S}^t = \begin{cases} [\hat{S}^t, \hat{S}^{t-1}, \dots, \hat{S}^{t_f+1}, S^t, S^{t-1}, \dots, S^{t-(h-1)}] & \text{if } t_f < t < t_f + h \\ [\hat{S}^t, \hat{S}^{t-1}, \dots, \hat{S}^{t-(h-1)}] & \text{if } t \geq t_f + h \end{cases} \quad (12)$$

여기서, S^* 는 \overline{S}^t 에 포함되어 있는 이동 상태들을 나타내며 \hat{S}^* 는 **NFIS**에 의해서 예측된 이동 상태들을 나타낸다.

(12)는 시간 t 에서의 이동 상태 벡터(\overline{S}^t)에 S^* 가 포함되는지의 여부에 따라서 서로 다른 형태의 이동 상태 벡터를 구성한다. 다음 이동 상태인 시간 $t+1$ 에서의 이동 상태를 예측하기 위해서 **NFIS**의 입력으로 \overline{S}^t 가 사용된다. **NFIS**에 의해서 예측된 이동 상태 \hat{S}^{t+1} 는 $t+2$ 에서의 이동 상태를 예측하기 위해 구성되는 시간 $t+1$ 의 이동 상태 벡터 \overline{S}^{t+1} 에 포함된다. 즉, 예측된 결과 값이 다음 예측을 위해서 **NFIS**의 입력 값으로 되먹임(feedback)된다. 따라서, $t+h$ 이전의 이동 상태 벡터들은 \overline{S}^t 에 포함된 이동 상태와 **NFIS**에 의해서 예측된 이동 상태가 혼용되어 구성되는 반면, $t+h$ 이후의 이동 상태 벡터들은 **NFIS**에 의해서 예측된 이동 상태들로만 구성되게 된다. 이러한 과정이 미래 시간 t_p 까지 반복되어 실패 후 미래 시간 t_p 에서의 이동 상태 \hat{S}^h 가 예측된다. 최종적으로 **NFIS**를 의해서 예측된 위치 정보는 잃어버린 사용자의 위치를 파악하기 위해 첫번째로 수행되는 페이징 셀로서 사용된다. 여기서, 예측된 위치 정보는 실패 시점에 사용자가 상주 가능성이 있는 셀을 나타낸다. 이때, 사용자의 상주 가능성이 있는 셀에 대한 정확성은 t_f 와 t_p 사이의 시간 간격에 의존한다. 이는 이러한 시간 간격이 크면 클수록 사용자는 이동성 데이터베이스의 실패 시간 t_f 에서의 상주 위치로부터 벗어나 움직일 가능성이 높아지기 때문이다. 그러나, 이동성 학습 및 예측 시스템이 실패 없는 연산 기간 동안 사용자의 이동 패턴을 정확히 학습하였다면, 예측된 위치 정보 또한 상주 가능성이 있는 셀로 높은 정확도를 갖고 예측할 것이다. 이러한 경우, 첫번째 페이징 이후 추가적인 페이징 과정 없이 예측된 셀에 대한 한번의 페이징만으로도 실패 후 사용자의 위치를 찾을 수 있다. 그러므로, 실패 후 사용자의 상주 위치에 대한 예측 정확도가 높아질수록 페이징 비용은 절감된다.

본 논문의 회복 기법은 HLR 실패 회복 과정과 VLR 실패 회복 과정으로 구성된다. 그림 6과 그림 7은 각각 HLR과 VLR의 실패 회복 과정에 대한 전반적인 흐름을 보여준다.

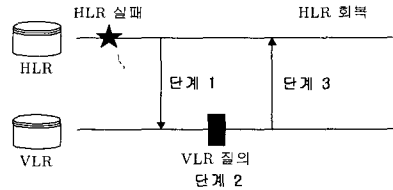


그림 6 HLR의 실패 회복 과정

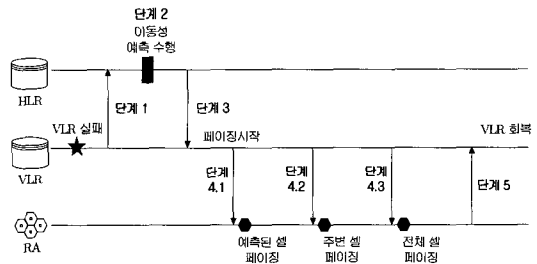


그림 7 VLR의 실패 회복 과정

HLR과 VLR의 실패 후 두 이동성 데이터베이스는 다음의 단계들을 각기 수행함으로써 회복된다.

이동성 예측에 의한 HLR 실패 회복

[단계 1] 실패된 HLR은 사용자의 잃어버린 위치 정보를 모으기 위해서 회복 개시 메시지(restoration initiation message)를 모든 VLR에 보낸다.

[단계 2] 회복 개시 메시지를 받은 각 VLR은 사용자의 위치 정보에 대한 질의를 수행한다.

[단계 3] 실패된 HLR은 모든 VLR로부터 정확한 위치 정보를 받고 이러한 위치 정보를 사용하여 실패된 HLR은 회복된다.

이동성 예측에 의한 VLR 실패 회복

[단계 1] 실패된 VLR은 사용자의 잃어버린 위치 정보를 모으기 위하여 HLR로 회복 개시 메시지를 보낸다.

[단계 2] 회복 개시 메시지를 받은 HLR은 요청된 사용자의 위치 정보를 이동성 학습 및 예측 시스템을 이용하여 예측한다.

[단계 3] 실패된 VLR은 HLR로부터 예측된 위치 정보를 받는다.

[단계 4] 예측된 위치 정보가 정확한 위치 정보인지 혹은 쓸모 없는 위치 정보인지를 검증하기 위하여 다음과 같은 단계를 수행한다.

[단계 4.1] 예측된 위치 정보에 대응되는 셀에 대해서 첫번째 페이지를 수행한다. 만약 사용자로부터 응답 메시지를 받는다면, 그 사용자에 대한 위치 정보를 재구성한다.

[단계 4.2] 일정한 시간 내에 응답 메시지가 없다면, 예측된 위치의 주변 셀에 대해서 두번째 페이지를 수행한다.

[단계 4.3] 주변 셀로의 페이지에 대해서도 어떠한 응답이 없다면, 예측된 위치 정보에 해당되는 RA 내의 모든 셀에 대해서 세번째 페이지를 수행한다.

[단계 5] 실패된 VLR은 페이지에 의해서 예측된 위치 정보에 대한 검증을 마치고 정확한 위치 정보를 받음으로써 실패된 VLR은 회복된다. 만약 단계 4의 연속적인 세번의 페이지에도 불구하고 사용자로부터 어떠한 응답이 없다면, 실패된 VLR은 사용자가 호를 시작할 때까지 기다린다.

실패된 HLR은 실패 후 VLR로의 질의를 통하여 HLR 내의 사용자의 잃어버린 위치 정보를 회복한다. VLR의 실패 회복 과정은 기존의 검사점 기법과 달리 사용자 이동성의 예측에 의해서 사용자의 잃어버린 위치 정보를 얻는다. 예측된 위치 정보의 정확성을 검증하기 위하여 최대 세 번까지의 페이지가 수행되는데, 이동성 학습과 예측 시스템이 사용자의 위치 정보에 대한 정확한 예측을 수행한다면, 첫번째 페이지만으로도 사용자의 위치 정보는 파악된다. 첫번째 페이지에 의해서 잃어버린 사용자의 위치가 파악되지 않으면, 예측된 셀에 대한 주변 셀의 페이지가 수행된다. 주변 셀에 대한 페이지에 의해서도 사용자의 위치가 파악되지 않으면, RA 내에 포함된 전체 셀을 페이지함으로써 사용자의 위치 정보는 파악된다.

이동성 예측은 사용자의 움직임에 대한 학습 정보를 기반으로 사용자가 미래에 어떤 셀에 상주할 지를 예측하는 것이다. 사용자의 이동성이 작다면 사용자는 예측된 셀에서 벗어날 확률이 적어지므로, 예측된 셀에 사용자가 상주할 가능성은 높아진다. 이때에는 첫번째 페이지에 의해서 사용자의 위치는 파악된다. 반면, 사용자의 이동성이 커질수록, 사용자는 예측된 셀 주변으로 이동할 확률이 많아지므로, 첫번째 페이지에 의한 사용자의 위치 파악 가능성은 줄어들게 되고, 두번째와 세번째 페이지가 수행될 가능성이 높아진다. 이것은 회복과정에서 필요로 하는 페이지 비용과 관련되는데, 이동성 학습과

예측 시스템이 사용자의 이동성에 상관없이 정확한 셀 예측을 수행한다면 회복되는 위치 정보가 올바른 회복 인지의 검증을 위해서 수행되는 페이지의 수를 줄임으로써 회복 과정에서 소요되는 비용을 줄일 수 있다.

검사점에 의한 회복 기법은 사용자의 위치 정보가 실패 없는 연산 기간 동안 안전한 저장소로 단순히 백업하여 실패 후 즉각적으로 회복시킨다. 이 회복 기법은 회복되는 위치 정보가 올바르지 못한 가능성을 내포하고 있고 이를 적절히 처리할 수 있는 과정을 가지고 있지 않다. 따라서 검사점에 의한 회복 기법의 복구율은 사용자의 이동성에 매우 민감하다. 반면, 제안하는 회복 기법은 잃어버린 위치 정보가 실패 시점에서 사용자 이동성 예측에 의해서 회복되기 때문에, 검사점에 의한 회복 기법과 달리 실패 후 상주할 만한 위치를 사용자의 이동성에 따라 능동적으로 파악한다. 아울러, 예측된 위치 정보는 실패 없는 연산 동안 사용자의 이동 패턴의 학습에 기반하여 유도되기 때문에, 제안하는 회복 기법은 사용자의 이동성에 상관없이 높은 예측 정확도를 가질 것으로 기대되며 이로 인하여 검사점 기법보다 훨씬 적은 페이지를 유도한다.

6. 성능 평가

이 장에서는 시뮬레이션을 통하여 본 논문에서 제안된 회복 기법의 성능 평가를 보여준다. 시뮬레이션의 수행에 앞서, 제안된 회복 기법의 성능 평가를 위해 사용될 사용자의 다양한 이동성 데이터의 추출이 필요하다. 따라서, 이 장에서는 이동성 데이터의 생성 방법을 먼저 기술하고, 이 방법에 의해서 생성된 이동성 데이터를 사용하여 제안된 회복 기법과 검사점에 의한 회복 기법 [10]의 성능을 회복 정보의 올바른 정보 파악을 위해 소용되는 페이지 비용 관점에서 서로 비교한다.

6.1 이동성 데이터 생성

실제 이동 환경에서, 사용자의 움직임 경향은 산, 강, 지형지물 등의 환경에 의한 요인, 사용자의 개인적인 이동 취향에 의한 요인 등에 의하여 다양한 이동 습성을 보이는 것이 일반적이다. 사용자의 이동성은 공간 차원에서 주어진 지리적 위치에서 다른 위치로 이동하는 행동으로 정의할 수 있으며, 사용자의 이동 습성에 영향을 받는다. 사용자의 이동성을 다룬 기존 연구들 [17, 18, 19]은 주로 랜덤워크 모델(random-walk model)에 기반하여 사용자의 이동성을 모델링하고자 하였으나, 랜덤워크 모델은 사용자의 이동 습성에 관한 측면을 고려하는데 어려움이 있다. 사용자의 이동 습성이 고려된 모델링 방법으로서 사용자의 이동 타입에 따라서 이동 패턴을 표

현할 수 있는 traveling demand 모델이 제안되어 왔다 [20]. 이 모델은 도시 지역에서 물류 수송을 표현하고 예측하기 위한 수송 계획과 트래픽 모델에 기반한다[21]. 본 논문에서는 traveling demand 모델을 이용하여 사용자의 이동 속도와 방향 관점에서 이동성 데이터를 추출한다.

이동 속도는 평균 이동 속도와 평균 이동 속도에서의 변화율로서 해석될 수 있다. 실제 상황에서 사용자가 평균 이동 속도로 이동한다면, 속도 변화율은 거의 0에 가까울 것이다. 반면, 사용자가 이동 속도에 대한 어떤 주기성 없이 이동한다면, 그들의 이동 경로에서 평균 이동 속도를 기준으로 무작위 변화율을 갖게 된다. 따라서, 본 논문에서는 시간 $t+1$ 에서의 이동 속도, v_{t+1} 을 다음과 같이 정의한다:

$$v_{t+1} = \bar{v}_t + \hat{v}_t \quad (13)$$

여기서, \bar{v}_t 는 시간 t 에서 평균 이동 속도를 나타낸다. 그리고, \hat{v}_t 는 시간 t 에서 이동 속도 변화율을 나타내며, 임의의 속도 범위 $[-\hat{v}_c, \hat{v}_c]$ 에서 균등 분포(uniform distribution)를 갖도록 선택된다.

한편, 사용자가 자신의 목적지를 향해 움직인다는 가정 하에, 사용자의 이동 방향은 목적위치를 향하는 주 이동 방향과 임의의 위치에서 사용자의 순간 이동 방향을 나타내는 현재 이동 방향에 의해서 얻어진다. 이와 같은 가정 하에 시간 t 에서의 현재 이동 방향으로부터 시간 $t+1$ 에서의 이동 방향 θ_{t+1} 는 다음과 같이 계산된다:

$$\theta_{t+1} = \bar{\theta}_t + \frac{2\pi}{d} \cdot k \quad (14)$$

여기서, $\bar{\theta}_t$ 는 시간 t 에서 목적지를 향하는 주 이동 방향을 나타내고, d 는 이동 방향의 개수를 나타낸다. 그리고, k 는 이동 방향의 인덱스를 나타내며 k 방향에 대한 확률 분포 $p_k(k \in \{0, 1, \dots, d-1\})$ 에 의해서 선택되도록 한다. (14)에서 k 방향에 대한 확률이 다른 이동 방향에 대한 확률보다 크다면, 사용자는 급변하는 방향 전환 없이 미리 정의한 목적지로 거의 일정하게 움직일 것이다. 반면, 모든 방향에 대한 확률이 같으면, 사용자는 불규칙적인 방향성을 갖고 목적지로 이동한다. 이 경우, 사용자는 목적지로 향하는 직선 방향에서 크게 벗어나는 이동 경로를 갖는다.

위에서 기술한 이동성 모델을 통하여 사용자는 이동 속도와 이동 방향으로 구성되는 이동성 타입 별로 분류된다. 본 논문에서는 각 이동성 타입에 대해서 이동성 데이터를 생성하고 생성된 데이터를 제안된 회복 기법의 성능 평가를 위해서 사용한다.

6.2 시뮬레이션 환경

본 논문에서의 사용자는 자신의 이동 특성에 따라 임의의 이동 경로를 따라 육각형 셀 환경의 임의의 셀로 움직인다고 가정한다. 육각형 셀 환경은 셀 반경이 500 m인 256개의 셀로 구성되어 있으며, 각 VLR은 동일한 셀 수를 갖고 있다고 가정한다. 이러한 육각형 셀 환경에서 사용자의 위치는 30일 동안 0.0835 시간 간격으로 측정되었다.

이동성 데이터를 추출하기 위해 본 논문에서는 이동 속도와 방향을 각각 세가지 종류로 나누어서 사용자를 전체 9가지 이동성 타입 별로 분류하였다. 표 1은 속도 타입(A, B, C)과 방향 타입(X, Y, Z)에 대한 사용자 분류와 파라미터를 보여준다. 표 1에서 보여준 사용자 타입 중에 AX 타입을 갖는 사용자는 항상 평균 속도와 방향을 갖고 임의의 목적지로 향해 움직인다. 반면, CZ 타입의 사용자는 자신의 이동 경로에서 무작위 속도와 방향을 보여준다.

시뮬레이션의 단순성을 위해서, 본 논문에서는 모든 시간 t 에 대해서 \bar{v}_t 와 d 를 각각 5 km/h와 다섯 방향으로 가정하였다. 따라서, 사용자는 이동 속도 변이(\hat{v}_c)와 이동 방향 확률벡터($\vec{p} = [p_0, p_1, \dots, p_{d-1}]$, $\sum_{k=0}^{d-1} p_k = 1$)의 값에 따라서 서로 다른 이동성을 보인다. 본 시뮬레이션의 수행을 위해 사용되는 파라미터들은 표 2와 같다.

표 1 사용자 타입

이동 속도			이동 방향			
타입	의미	값	타입	의미	값	
A	일 정	(5, 25)	X	일 정	[0.0 0.1 0.8 0.1 0.0]	
B	중 간	(10, 40)	Y	중 간	[0.1 0.2 0.4 0.2 0.1]	
C	무작위	(20, 60)	Z	무작위	[0.2 0.2 0.2 0.2 0.2]	

표 2 시뮬레이션을 위한 파라미터

파라미터	값
사용자의 이동성 데이터 개수(days)	30
현재와 과거의 이동 상태 개수(h)	4
이동 상태의 변경 주기(τ)	0.0835
클러스터의 반경(r)	0.3
학습률(η)	0.05
이동 상태에 대한 학습 횟수	50
퍼지 규칙의 나이를 조정하는 파라미터(α)	0.1
불필요한 퍼지 규칙의 제거를 위한 파라미터(β)	0.005

30일 동안 사용자 타입 별로 추출된 이동성 데이터를 사용하여 제안된 회복 기법의 성능을 평가한다. 실패 없는 연산 동안, *NFIS*는 이들 이동성 데이터를 사용하여 사용자의 이동 패턴을 학습하며, 이동성 데이터베이스의 실패는 30일 이후에 발생한다고 가정하였다. 실패 후 시간이 경과함에 따라 *NFIS*에 의해서 예측된 셀과 사용자의 실제 상주 셀과의 적중률을 통하여 성능 평가를 수행하며, 검사점에 의한 회복 기법의 적중률과 비교하여 제안된 회복 기법 기법의 우수성을 보일 것이다.

검사점에 의한 회복 기법의 성능은 사용자의 이동성에 따라서 좌우된다. 실패된 데이터 베이스가 안정한 저장소로부터 회복되자마자, 올바른 회복인지의 판별을 위해 회복된 위치에 대해서 페이지가 수행되는데, 첫번째 페이지에 의해서 잃어버린 사용자가 발견되지 않으면 위치 영역 내의 전체 셀에 대한 페이지가 수행되어야 한다. 전체 셀에 대한 페이지의 수행은 회복되는 정보가 쓸모 없게 되었을 경우를 의미하며 사용자가 검사점이 취해진 시점에서 상주했던 위치에서 벗어났기 때문에 수행된다. 따라서, 검사점에 의한 회복 기법의 성능은 검사점으로부터 회복된 위치 정보가 실패 후 사용자가 상주하는 위치 정보와의 일치 정도에 의해서 결정된다. [10]에서 이동성 데이터베이스가 담당하고 있는 임의의 위치 영역에 사용자의 상주 확률이 평균 $1/\lambda_m$ 을 갖는 지수분포를 따른다는 가정 하에 검사점에 의한 회복 기법의 성능을 분석하였다. 여기서, λ_m 은 사용자가 RA에서 벗어나는 평균 확률이다. 이러한 가정 하에 시간 $(0, T_V)$ 동안에 사용자가 임의의 위치 영역에 진입하고 진입한 위치에 대해서 검사점을 취하며 실패 후 T_V 시간에 검사점을 취한 위치를 벗어날 확률 P_{T_V} 를 다음과 같이 유도하였다:

$$P_{T_V} = \frac{1}{T_V} \int_0^{T_V} e^{-\lambda_m(T_V-s)} ds \quad (15)$$

$$= \frac{1 - e^{-\lambda_m \cdot T_V}}{T_V \cdot \lambda_m}$$

여기서, T_V 는 평균 검사점 간격을 나타낸다. (15)를 사용하여 검사점 기법에 기반한 회복 기법에서 사용자가 실패 후 검사점을 취한 위치에서 벗어나지 않을 확률을 유도할 수 있다. 이 확률 값이 낮아지면, 실패 후 잃어버린 사용자의 위치를 찾기 위해서 소용되는 페이지 비용은 점점 증가한다. 따라서, 본 논문의 시뮬레이션에서는 제안된 기법의 성능 비교를 위해서 (15)를 검사점에 의한 회복 기법의 성능으로 사용한다. 본 논문에서는 λ_m 을 0.4로 가정하여 검사점에 의한 회복 기법의

성능을 평가한다.

제안된 회복 기법의 성능은 실패 후 *NFIS*에 의해서 예측된 결과와 실제 위치와의 정확성 여부에 의해서 결정된다. 제안된 회복 기법이 실패 후 사용자가 상주할 만한 위치를 정확히 예측할 수 있으면, 잃어버린 사용자를 찾기 위해서 필요로 하는 페이지가 단 한번만 수행되어 위치 영역에 대한 전체 셀의 페이지에 비해서 페이지 비용을 상당히 감소시킬 수 있다. 따라서, 제안된 회복 기법의 성능 분석을 위해서 본 논문에서는 이동성 데이터베이스의 실패 후 시간이 경과함에 따라 정확히 회복되는 위치 정보의 수를 평가한다. 성능 평가의 기준으로 사용되는 정확한 셀 적중률(exact cell hit ratio; 이하 *ECHR*)과 인접한 셀 적중률(adjacent cell hit ratio; 이하 *ACHR*)을 다음과 같이 정의한다.

$$ECHR = \frac{\text{한번의 페이지로 올바르게 복구되는 사용자 위치정보의 수}}{\text{잃어버린 사용자 위치 정보의 수}} \times 100 \quad (16)$$

$$ACHR = \frac{\text{두 번의 페이지로 올바르게 복구되는 사용자 위치정보의 수}}{\text{잃어버린 사용자 위치 정보의 수}} \times 100 \quad (16)$$

여기서, *ECHR*은 실패 후 경과한 임의의 미래 시간에서 예측된 셀에 대해서 한번의 페이지만으로 정확히 복구되는 비율을 나타내며, *ACHR*은 예측된 셀의 주위 셀을 페이지하여 복구되는 비율을 나타낸다.

6.3 시뮬레이션 결과

본 절에서는 앞서 기술한 시뮬레이션 환경 하에서 본 논문의 회복 기법에 대한 시뮬레이션 결과를 보여주고 검사점에 기반한 회복 기법과 성능을 비교한다. 그림 8은 이동성 데이터베이스의 실패 후 경과한 시간에 따른 모든 사용자 타입에 대한 평균 복구율을 보여준다. 그림 8에서 사선은 검사점에 기반한 회복 기법의 평균 복구율을 나타내며, 점선과 점쇄선은 각각 제안하는 회복 기법에서의 평균 *ECHR*과 *ACHR*을 나타낸다. 그림 8에서 볼 수 있듯이, 검사점에 기반한 회복 기법은 본 논문의 회복 기법에 비해서 시간의 경과함에 따라서 급격한 감소를 보이며, 더욱이 실패 후 경과한 초기 시간부터 급격한 감소 추세를 보인다. 반면, 본 논문의 회복 기법은 실패 후 경과 시간이 50분에서 70분까지 비교적 높은 평균 복구율을 보여준다. *ECHR*이 다소 낮은 평균 복구율을 보여주지만, 예측된 셀의 인접 셀까지를 적중률로 계산하는 *ACHR*은 90% 이상의 높은 평균 복구율을 보여준다. 그림 8의 결과로부터 검사점에 기반한 회복 기법은 실패 후 경과 시간에 매우 민감한 성능을 보

여움을 알 수 있다. 반면, 본 논문의 회복 기법은 실패 후 많은 시간이 경과했음에도 불구하고 비교적 높은 평균 복구율을 보여주는데, 이는 본 논문의 회복 기법이 이동성 데이터베이스의 실패 후에 상주할 위치에 대한 예측 정보를 회복 과정 중에 사용하였기 때문이다. 게다가, 본 논문의 회복 기법은 적은 수의 페이지만으로 효과적으로 잃어버린 위치 정보를 회복시킬 수 있으므로 페이지 비용의 획기적인 절감 효과를 얻을 수 있다. 이에 반해, 검사점에 기반한 회복 기법은 첫번째 페이지에 의해서 잃어버린 사용자를 찾지 못한 경우에 등록 영역에 대한 전체 페이지가 수행되어야 한다. 실제 시스템에서 하나의 등록 영역 내에는 수 백 개의 셀들로 이루어져 있으므로 검사점에 기반한 회복 기법은 본 논문의 회복 기법에 비해서 잃어버린 사용자를 찾기 위해서 더 많은 셀을 페이지 해야만 하고 이로 인하여 페이지 비용의 증가를 초래한다.

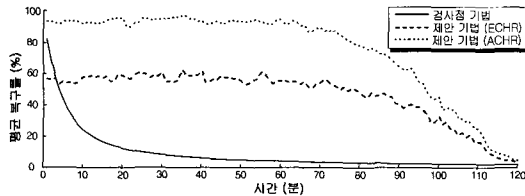
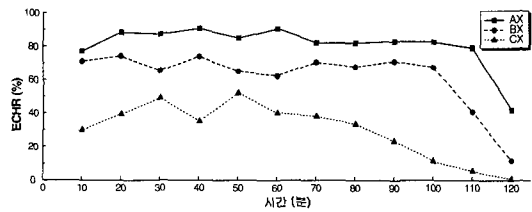


그림 8 실패 후 경과 시간에 따른 평균 복구율

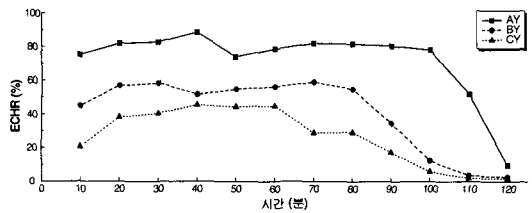
다음에서 사용자의 움직임이 본 논문의 회복 기법에 어떤 영향을 끼치는지를 살펴본다. 그림 9는 실패 후 경과한 시간에 따른 9개의 사용자 타입의 ECHR을 보여준다. 그림 9에서 AX 사용자 유형은 다른 사용자 유형에 비해서 높은 복구율을 보여주는데, 이는 AX 사용자 유형이 비교적 규칙적인 이동 패턴을 가지므로 실패 후에 사용자가 상주할 위치 또한 일정하기 때문이다. 이들 사용자는 실패 후 경과 시간이 거의 100분이 지났음에도 불구하고 80%에서 90%까지의 ECHR을 보여준다. 반면, 불규칙적인 이동 패턴을 갖는 CZ 사용자 유형은 낮은 복구율을 보여준다. 이들 사용자 유형의 ECHR이 다른 사용자 유형의 ECHR보다 그다지 높지는 않지만, 실패 후 초기 단계에서 갑작스런 감소를 보이지 않고 60분까지 실패가 지속될 때까지 일정한 ECHR을 보여 주었다.

그림 9에서 보여준 모든 사용자 유형의 ECHR을 분석해볼 때, 사용자의 이동 속도와 방향의 불규칙성이 증가함에 따라 이동성 데이터베이스의 실패 후 회복되는 위치정보의 복구율은 점점 낮아지지만, 모든 사용자 유

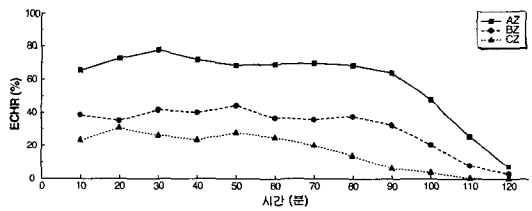
형에 대해서 실패의 초기 시간부터 급격한 복구율의 감소는 보이지는 않는다. 이것은 사용자의 이동성에 상관 없이 실패의 초기 단계에서부터 균등한 페이지 비용으로 잃어버린 사용자의 위치 정보를 회복시킬 수 있음을 의미한다. 따라서, 본 논문에서 제안하는 이동성 예측에 의한 회복 기법은 검사점에 의한 회복 기법에 비해서 훨씬 적은 셀만을 페이지하여 위치 정보를 회복할 수 있으며, 더욱이 실패의 초기 단계에서부터 균등한 페이지 비용으로 회복시킬 수 있다. 특히, 비교적 일정한 이동성을 보이는 사용자들이 실패 후 시간이 경과함에 따라 복잡한 이동성을 갖는 사용자보다 훨씬 좋은 복구율을 갖음을 알 수 있다.



(a) 이동 방향이 일정한 경우



(b) 이동 방향이 중간인 경우



(c) 이동 방향이 무작위인 경우

그림 9 모든 사용자 타입에 대한 ECHR

7. 결론

본 논문에서는 이동 통신 시스템에서 이동성 데이터베이스의 실패 회복을 위해서 이동성 학습과 예측에 기반한 회복 기법을 제안하였다. 제안된 회복 기법은 실패 없는 연산 기간동안 사용자의 새로운 위치가 등록될 때마다, 이들 사용자의 이동 패턴을 뉴로-퍼지 추론 시스

템을 사용하여 학습한다. 이동성 데이터베이스에 실패가 발생하였을 때, 뉴로-퍼지 추론 시스템에 의한 이동성 예측 수행을 통하여 얻어진 예측 정보를 실패 후 잃어버린 사용자가 상주할 가능성이 있는 위치를 결정하기 위해서 사용한다. 이동성 데이터베이스의 실패를 회복시키기 위한 이동성 학습과 예측의 사용은 사용자의 이동성을 회복 과정에 효율적으로 반영함으로써 검사점에 기반한 기존의 회복 기법과 비교하여 훨씬 적은 페이지 비용으로 실패 후 잃어버린 사용자의 위치정보를 회복한다.

시뮬레이션을 통한 제안된 회복 기법의 성능 분석은 이동성 학습과 예측에 의해서 비교적 높은 회복 복구율을 유도함을 보여주었다. 검사점에 의한 회복 기법은 검사점 간격이 넓어짐에 따라 복구율은 점점 낮아지는데 반해, 이동성 학습과 예측에 의한 본 논문의 회복 기법은 이동성 데이터베이스의 실패 후 상당히 오랜 시간이 지남에도 불구하고 비교적 높은 복구율을 유지하였다. 또한, 사용자 이동성의 복잡성의 영향을 보인 시뮬레이션에서는 제안된 회복 기법이 사용자의 이동 속도와 방향의 변이에 의해서 다소 성능 상에 영향을 끼치었지만, 비교적 일정하게 높은 복구율을 갖고 잃어버린 사용자의 위치 정보를 회복시킬 수 있음을 보여주었다.

본 논문의 회복 과정에 사용되는 뉴로-퍼지 추론 시스템은 기존의 검사점 기법에 의한 회복 기법에서의 검사점의 역할을 이동성 학습과 예측으로 충분히 대신하였으며 성능 분석을 통한 시뮬레이션 결과에서는 높은 회복 복구율을 보여 주었다. 아울러, 제안한 회복 기법은 검사점에 의한 회복 기법의 실패 없는 연산 기간 동안 요구되는 검사점을 위한 저장 공간과 백업 과정을 수행할 필요가 없으므로 실패 없는 연산 기간 동안에 소용되는 비용을 줄일 수 있다. 따라서, 본 논문에서 제안한 이동성 학습과 예측에 기반한 회복 기법은 검사점에 의한 회복 기법에 비해서 훨씬 감소된 지연과 적은 비용을 갖고 비교적 높은 회복 복구율을 통하여 실패된 이동성 데이터베이스를 회복시킬 수 있음을 알 수 있다.

참 고 문 헌

- [1] T. Imielinski and H.F. Korth, *Mobile Computing*, Kluwer Academic Publishers, 1996.
- [2] E. Pitoura and G. Samaras, *Data Management for Mobile Computing*, Kluwer Academic Publishers, 1998.
- [3] I.F. Akyildiz and J.S.M. Ho, "A mobile user location update and paging mechanism under delay constraints," *Computer Communication Review*, vol. 25, no. 4, pp. 244-255, Oct. 1995.
- [4] B. Awerbuch and D. Peleg, "Online tracking of mobile users," *Journal of ACM*, vol. 42, no. 5, pp. 1021-1058, Sep. 1995.
- [5] A. Bar-Noy and I. Kessler, "Tracking mobile users in wireless communication networks," *IEEE Trans. on Information Theory*, vol. 39, no. 6, pp. 1877-1886, Nov. 1995.
- [6] S. Tabbane, "An alternative strategy for location tracking," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 5, pp. 880-892, June 1995.
- [7] EIA/TIA, "Cellular radio-telecommunications inter-systems operations: Automatic roaming," *Technical Report IS-41.3-B*, EIA/TIA, 1991.
- [8] ETSI/TC, "Restoration procedures, version 4.2.0," *Technical Report Recommendation GSM 03.07*, ETSI/TC, 1993.
- [9] Y.-B. Lin, "Failure restoration of mobility databases for personal communication networks," *ACM-Baltzer Journal of Wireless Networks*, vol. 1, no. 3, pp. 365-372, 1995.
- [10] T.-P. Wang, C.-C. Tseng, and W.-K. Chou, "An aggressive approach to failure restoration of PCS mobility databases," *Mobile Computing and Communication Review*, vol. 1, no. 3, pp. 21-28, Sep. 1997.
- [11] G. Liu and G. Maguire Jr., "A class of mobile motion prediction algorithms for wireless mobile computing and communications," *Mobile Networks and Applications*, vol. 1, pp. 113-121, 1996.
- [12] T. Liu, P. Bahl, and I. Chlamtac, "Mobility modeling, location tracking, and trajectory prediction in wireless ATM networks," *IEEE Journal on Selected Areas in Communications*, vol. 16, no. 6, pp. 922-936, Aug. 1998.
- [13] J. Chan and A. Seneviratne, "A practical user mobility prediction algorithm for supporting adaptive QoS in wireless networks," *Proc. of IEEE Int'l Conf. on Networks (ICON '99)*, pp. 104-111, Sep. 1999.
- [14] 길준민, 박찬열, 양권우, 한연희, 황중선, "이동 컴퓨팅 시스템에서 뉴로-퍼지 추론 시스템을 이용한 지능적 이동성 예측", *한국정보과학회 논문지(A)*, 제26권, 제4호, pp. 472-487, 1999.
- [15] J.-M. Gil, C.Y. Park, Y.-H. Han, C.-S. Hwang, and Y.-S. Jeong, "Simulation of a Mobility Prediction Scheme by Neuro-Fuzzy Theory in Mobile Computing," *Simulation Journal (Special Issue on Mobile and Wireless Communications and Information Processing)*, vol. 75, no. 1, pp. 6-17, July 2000.

- [16] J.-S. R. Jang, C.-T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing*, Prentice Hall, 1997.
- [17] R. Thomas, H. Gilbert, and G. Mazziotto, "Influence of the movement of mobile station on the performance of the radio cellular network," *Proc. of 3rd Nordic Seminar*, Sep. 1988.
- [18] D. Hong and S.S. Rappaport, "Traffic model and performance analysis for cellular mobile radio telephone systems with prioritized and non-prioritized handoff procedure," *IEEE Trans. on Vehicular Technology*, vol. 35, no. 3, pp. 77-91, Aug. 1986.
- [19] D. Lam, D.C. Cox, and J. Widom, "Teletraffic modeling for personal communication services," *IEEE Comm. Mag. Special Section on Teletraffic Modeling*, vol. 35, no. 2, pp. 79-87, Feb. 1997.
- [20] J. Scourias and T. Kunz, "A dynamic individualized location management algorithm," *Proc. of the 8th IEEE Int'l Symposium on Personal, Indoor, and Mobile Radio Communications*, pp. 1004-1008, Sep. 1997.
- [21] N. Oppenheim, *Urban Travel Modeling*, A Wiley-Interscience Publication, 1994.



길 준 민

1994년 고려대학교 자연과학대학 전산학과 졸업 (이학사). 1996년 고려대학교 대학원 전산학과 졸업 (이학석사). 2000년 고려대학교 대학원 컴퓨터학과 졸업 (이학박사). 1998년 ~ 현재 고려대학교 기초과학연구소 연구원. 2001년 ~ 현재 일리노이대학교(시카고) 전산학과 Post-Doc. 관심분야는 이동 컴퓨팅, 무선 네트워크, 이동 객체 데이터베이스, 분산 시스템, 소프트 컴퓨팅 등임

황 종 선

정보과학회논문지 : 정보통신
제 29 권 제 3 호 참조



정 영 식

1987년 고려대학교 수학과(이학사). 1989년 고려대학교 전산학과(이학석사). 1993년 고려대학교 전산학과(이학박사). 1998년 미시간주립대학교 교환교수. 1993년 ~ 현재 원광대학교 컴퓨터 및 정보통신 공학부 부교수. 관심분야는 병렬분산시스템, 컴퓨터시뮬레이션, 그리드 컴퓨팅, 이동컴퓨팅