

외래환자 진료 프로세스를 위한 인터넷 기반 Generic 시뮬레이션 모델*

구정인** · 이기호** · 이종성** · 이 홍** · 김병인**†

Internet-based Generic Simulation Model for Outpatient Clinics

Jeongin Koo** · Giho Lee** · Jongsung Lee** · Hong Li** · Byung-In Kim**

■ Abstract ■

This paper introduces an Internet based generic simulator for outpatient clinics, named PIOS (Postech Internet based Outpatient Simulator). Unlike other simulation tools, PIOS is operated on the Internet, is straightforward to use for outpatient process analysis, and is open to public via <http://logistics.postech.ac.kr/pios> with free of charge. Therefore, non simulation expert users such as hospital administrators and managers can easily access to PIOS and conduct simulation study by themselves. In this paper, we describe the main components and programming logic of PIOS, and demonstrate its validity by comparing the output results of PIOS and a commercial software package.

Keyword : Healthcare, Generic Simulation, Internet-Based Simulation, Outpatient Clinics

1. 서 론

의료서비스 산업은 “의료서비스”를 제공하고 소

비자로부터 대가를 받는 과정에 관련된 산업을 지칭하며 의료서비스는 질병의 치료, 간호, 예방, 관리 및 재활을 주체로 하는 의료 서비스와 건강유지 및

논문접수일 : 2012년 09월 25일 논문게재확정일 : 2012년 12월 05일

논문수정일(1차 : 2012년 11월 27일)

* 이 논문은 2011년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 2010-0015960).

** 포항공과대학교 산업경영공학과

† 교신저자

증진을 포함한 포괄적 의료서비스를 포함한다. 이러한 의료서비스는 서비스 산업에서 중요한 부분을 차지하고 있으며 인구 고령화와 서비스의 질적 향상에 대한 인식증가와 같은 요소들을 고려할 때 앞으로도 그 비중이 증가할 것이다. 그러나 국내 서비스 산업의 생산성은 OECD 선진국 수준에 비해 매우 낙후되어 있으며, 의료서비스업에서도 비슷한 상황이다[3]. 하지만 국내 의료서비스 산업을 개선하기 위한 혁신활동은 의료법의 실정이나 의료기관의 영세성 등으로 인해서 미미한 수준이다[4]. 이는 미국 등의 선진국에서 이미 1950년대 후반부터 의료서비스 분야에 경영과학 이론 및 해법을 적용하려는 시도를 했고 1980년대부터는 실제문제들에 적용하고 있는 것과 유럽 경영과학회의 의료분야 활동에 대조되는 부분이다[4].

국내 의료서비스 개선을 언제까지나 미미한 수준으로 둘 수는 없다. 이미 선진국에서 성공적으로 적용된 경영과학 기법들은 신속히 국내 의료산업에도 적용할 수 있어야 하며 더 나아가 우리 나라의 실정에 맞는 방법론들을 개발하여 적용하여야 한다. 국내 의료계의 상황 또한 이러한 필요를 요구하고 있다. 정부는 의료법을 개정하여 의료법인의 부대사업으로 병원경영지원 사업을 허용할 계획인데, 이를 통해 의료법인들이 병원경영지원회사(MSO, Management Service Organization) 설립이 가능해질 것이다[9]. 병원경영지원회사의 형태의 하나로 경영지원형 MSO가 있는데 이는 의약품, 의료기기 등의 구매 대행, 의료시설 등차원 공유, 인력관리 등 경영활동의 아웃소싱과 진료연계를 통해 네트워크에 포함된 병원들의 경쟁력 강화를 추구하는 형태이다. 이러한 자원이나 인력관리와 같은 경영활동들은 경영과학적 해법의 기회와 필요성을 더 증가시킬 것이다.

의료서비스 분야의 경쟁력은 크게 의술의 경쟁력과 의술을 효과적으로 적용할 수 있도록 하는 시스템의 경쟁력으로 나눌 수 있다. 본 연구는 그 중 시스템의 경쟁력을 향상시키는 범주에 속하는 것으로 병원 프로세스의 가장 기본이 되는 외래환자 진료

프로세스 진단에 관한 것이다. 외래 서비스는 의료 서비스 전달의 대표적인 활동 중의 하나로 보건복지부에서 발간한 『2011 보건복지통계연보』에 따르면 2009년 기능별 국민의료비 구성비에서 31.3%의 비중을 차지하여 입원 30.4%, 의약품 22.5% 보다 높게 나타났다[5]. 따라서 외래 환자 흐름을 효과적으로 관리하는 것은 의료 서비스의 질에 큰 영향을 끼친다. 이와 같은 이유로 국내의 많은 병원들이 외래 환자의 흐름을 효과적으로 하기 위한 방법을 찾고 있다[7]. 이러한 문제에 적합한 방법론이 시물레이션이다. 하지만 시물레이션을 정확하게 사용하기 위해서는 소정의 교육과정을 이수해야 하고, 프로그래밍을 통해 시물레이션 모델을 개발하는데 많은 시간이 소요되며, 일반적으로 상용 소프트웨어를 구입해야 한다. 그러나 국내 의료기관의 영세성과 인력 구조를 볼 때 시물레이션 프로젝트를 통해 외래 프로세스를 개선할 수 있는 능력과 여유가 되는 기관은 많지 않다. 이러한 국내의 문제를 해결하고자 본 연구에서는 의료기관에서 별도의 시물레이션 소프트웨어 구매 없이 인터넷을 통해 복잡한 프로그래밍을 하지 않고 쉽게 외래 환자 프로세스 시물레이션을 수행해 볼 수 있도록 인터넷 기반 외래환자 진료 프로세스 진단용 Generic 시물레이터인 PIOS (POSTECH Internet-based Outpatient Simulator)를 개발하였다.

PIOS에서 적용된 것과 같은 generic 시물레이션 모델은 시물레이션 코드를 직접 수정하지 않고도 구조적 유사성을 가진 특정 도메인의 여러 시물레이션 문제에 적용이 가능한 시물레이션 모델을 뜻한다[21]. Pidd[21]는 데이터 기반의 generic 시물레이터의 특징으로 미리 프로그래밍 된 시물레이션 모델, 넓은 범위에 적합한 모델, 사용자의 프로그래밍이 불필요할 것, 사용자로부터 정보를 입력받을 것을 제시하였다. Generic 시물레이션 모델의 유용성은 많은 연구들을 통해서 이미 제조업에서 검증받았다. Mackulak et al.[18]의 generic 시물레이션 모델은 6주 이상 소요되던 시물레이션 제작 및 분석시간을 1주 이내로 줄일 수 있었으며 Steele et

al.[25]은 NASA의 재사용 가능한 발사대용 차량에 관한 generic 시뮬레이션 모델을 개발하였다. Kim et al.[14]과 Wy et al.[27]도 반도체 라인 및 일반 제품의 조립라인 시뮬레이션을 위한 generic 시뮬레이터를 개발하여 시뮬레이션 모델의 개발기간을 크게 단축하였다. 이 외에도 제조업 분야의 generic 시뮬레이션에 관한 대표적인 연구로 Lee and McLean [16], Mathewson[19], Aytug and Dogan[11], Lee et al.[15], Son et al.[24]이 있다.

의료 서비스 분야에서 시뮬레이션을 이용하여 프로세스를 개선한 사례는 권방현[2], 전희선 등[8], 최재혁 등[10], Lowery[17], Harper et al.[12], Rohleder et al.[23], Wijewickrama[26] 등을 포함하여 다수 찾아 볼 수 있다. Jun et al.[13]은 환자 흐름 개선 및 리소스 할당 부문에 시뮬레이션이 적용된 문헌 조사를 실시하였다. 이들은 병원에서 환자 대기 시간을 줄이기 위해 시뮬레이션을 이용하여 모델링 한 후 자원을 추가하거나 예약 시간을 변경하는 등 시스템을 변화 시킬 때 대기 시간 단축에 대해 연구하였으며 환자의 대기 시간을 단축시킬 수 있는 프로세스를 제안하는데 시뮬레이션이 효과적인 도구임을 보였다. 그러나 위 연구들은 특정 병원에서 시스템을 개선하려는 노력이 대부분이며 다양한 형태의 병원에 대해 시뮬레이션을 수행할 수 있는 generic 시뮬레이션에 대한 연구는 많지 않다. 영국의 Paul and Kuljis[20]는 영국의 작은 클리닉을 대상으로 시뮬레이션을 할 수 있는 도구로 CLINSIM이라는 시뮬레이션 시스템을 개발하였지만 시뮬레이션에서 허용하는 클리닉의 크기가 매우 제한적이라는 한계가 있다. 위주영 등[6]과 구정인 등[1]은 외래 환자 진료 시 필요한 generic 시뮬레이션 프레임워크를 제안하였고 실제 프로세스가 서로 다른 병원에 적용하여 프레임워크가 유효함을 보였지만 generic 시뮬레이션 모델의 작성에 상용소프트웨어를 사용하였기 때문에 영세한 병원이 많은 국내특성상 시뮬레이션 모델에 대한 병원관리자들의 접근성이 떨어지는 한계가 있다.

PIOS는 자바로 개발되어 인터넷을 통해서 <http://>

logistics.postech.ac.kr/pios에 공개되어 있다. 웹 기반으로 개발된 시뮬레이션 도구는 찾아보기 어렵다. Raikundalia et al.[22]이 개발한 웹 기반 병원운영 시뮬레이션 도구가 있지만 Simul8 소프트웨어를 기반으로 개발하였기 때문에 무료로 공개되어 있지 않으며 PIOS와는 달리 generic 시뮬레이션 모델은 아니다. PIOS의 개발목적 중의 하나는 시뮬레이션을 잘 알지 못하는 병원 관리자와 같은 사용자도 충분히 자신의 병원에 대한 시뮬레이션을 해 볼 수 있도록 하는 것이다. 사용자가 별도의 추가적인 시뮬레이션 프로그래밍 없이 웹 브라우저를 통해서 외래환자의 예약, 접수, 예비진료, 진료, 검사, 재진료, 처방, 수납, 병원 떠남의 과정을 정형화된 형식에 따른 데이터 입력만으로 시뮬레이션을 수행할 수 있고 시뮬레이션 결과 또한 사용자들이 이해할 수 있는 수준으로 자동으로 생성된다. 또한 사용자들이 자신이 원하는 병원의 환경을 시뮬레이션 모델로 쉽게 구축하고 시뮬레이션의 동작을 확인할 수 있도록 그래픽 화면을 제공하고 있다. 특히 PIOS가 기존의 모델들과 다른 점은 영세한 병원관리자들도 쉽게 접근할 수 있도록 무료로 공개되어 사용할 수 있도록 개발된 점이다.

본 논문의 나머지 부분은 다음과 같이 구성된다. 제 2장에서는 PIOS의 구성요소와 로직을 설명한다. 제 3장에서는 PIOS를 이용하여 만든 시뮬레이션 모델을 상용 시뮬레이션 소프트웨어인 AutoMod로 작성한 시뮬레이션 모델과의 비교분석을 통하여 로직의 구동 및 입력값 대비 결과 값들의 유효성을 검증하고 제 4장에서 결론을 맺는다.

2. PIOS의 구성요소 및 로직

2.1 PIOS의 구성요소

본 장에서는 PIOS의 구성요소들을 살펴보고 각 구성요소들 간의 로직의 흐름에 대해서 기술한다. PIOS는 자바 애플릿 형태로 개발되었으며 구성요소로 크게 엘리먼트(Element), 엔티티(Entity), 링크(Link),

〈표 1〉 PIOS의 구성 클래스

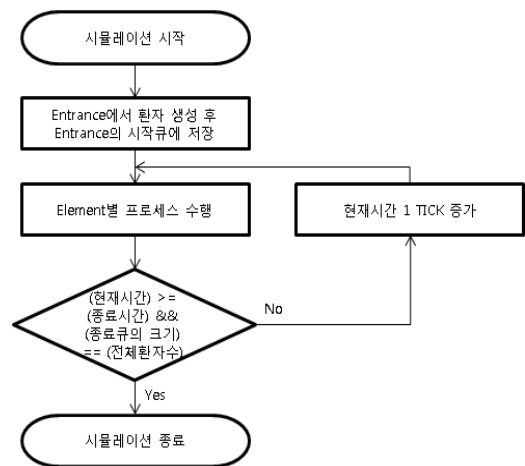
| 클래스이름 | 내용 |
|-------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 엘리먼트 | 추상클래스로 최상위 클래스 역할을 하며 환자를 제외한 나머지 클래스는 이를 상속받아 구성된다. 직접 오브젝트를 생성하는데 사용되지는 않는다. |
| 엔티티 | 병원 내에서 각 프로세스(접수, 검진 등)를 나타내기 위해서 사용되며 각 엔티티들은 큐(Queue)를 가지고 있다. 시뮬레이션 시작 시 도착 환자는 엔티티의 하위 클래스 중의 하나인 Entrance의 도착환자 큐에 환자 도착 정보에 따라 발생하여 순서대로 들어간다. 이후 큐에서 환자를 꺼내는 방식으로 시뮬레이션이 진행된다. 또한 환자가 병원을 떠나는 것도 이 클래스의 오브젝트로 구성되며 떠나는 환자는 내부 큐에 순서대로 저장된다. |
| 링크 | 병원 내 이동 경로를 나타낸다. 논리적 처리를 위해서 내부적으로 큐를 가지며 경로를 이동하는 환자는 해당 오브젝트의 내부 큐에 순서에 따라 존재한다. |
| 환자 | 병원 내의 환자를 나타내는 클래스로 자신이 받아야 하는 서비스를 가진다. |
| 크로스 | 병원 내 갈림길과 같이 이동 경로가 교차하는 지점을 나타낸다. 엔티티를 상속받으며 시뮬레이터에서 병원 내부를 묘사하고 내부 그림을 그리는데 사용된다. |

환자, 크로스(Cross)와 같이 5가지의 클래스로 이루어져 있다. <표 1>은 각 클래스와 그 역할을 기술한다.

PIOS의 시뮬레이션 진행은 각 단위 시간 간격(틱)으로 엔티티와 같은 오브젝트들의 상태를 반복적으로 체크하면서 진행한다. 1틱의 기본값은 1초(시뮬레이션상의 시간)로 정해져있다. 전체 시뮬레이션 순서는 [그림 1]과 같다. 시뮬레이션이 시작되기 전에 각 환자는 병원에 도착한 시간이나 진료를 받아야 할 프로세스의 순서와 같은 환자 정보에 따라서 생성이 되어 엔티티 클래스로 선언된 Entrance의 시작 큐에 저장되며 시뮬레이션이 시작되면 각 엘리먼트 별로 자신의 프로세스를 수행한다. 엘리먼트는 엔티티(Entrance 포함), 링크, 크로스로 나누어지며 각각 엘리먼트들이 시뮬레이션에서 수행하는 작업은 [그림 2]~[그림 4]와 같다. 크로스의 경우 실제 시뮬레이션 내에서 논리적으로 하는 일은 존재하지 않으며 시뮬레이션 애니메이션을 수행하고 갈림길과 같이 병원의 물리적인 공간을 표현하기 위해 사용된다. 그리고 각 엘리먼트 별로 수행해야 할 프로세스가 완료되면 시뮬레이션의 종료조건을 비교하고 종료조건을 만족한다면 시뮬레이션을 종료하며 만족하지 않는다면 시뮬레이션의 시간이 1틱만큼 증가한다.

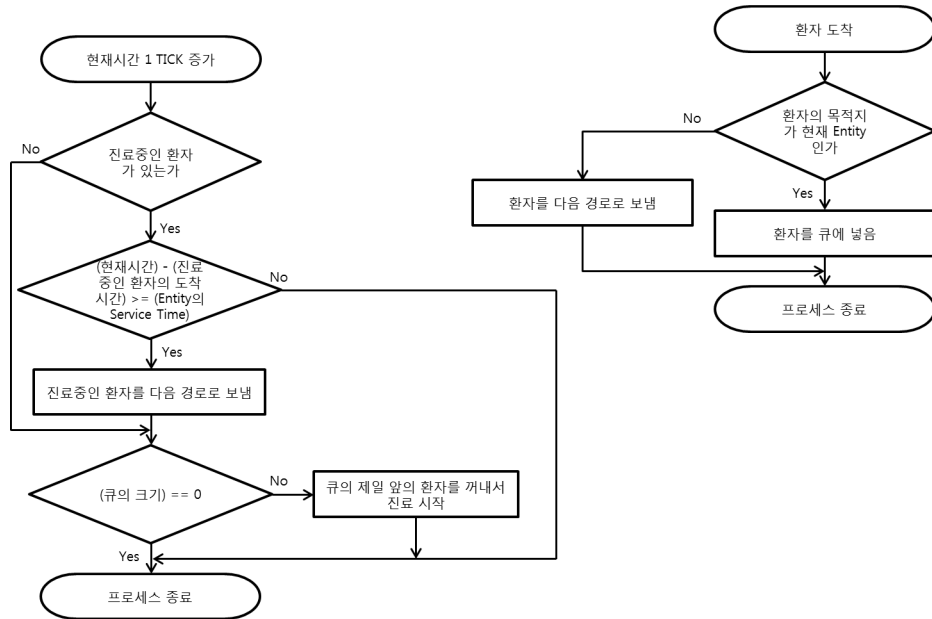
엔티티는 병원 내에서 각 진료 프로세스(접수, 진료 등)를 표현하기 위해 사용되는 클래스이며 환자

가 해당 엔티티에 도착할 때와 시뮬레이션이 1틱 증가 할 때 [그림 2]와 같은 작업을 수행한다. 시뮬레이션 시간이 1틱 증가했을 때, 엔티티는 현재 자신이 서비스 중인 환자가 있으면 진료 서비스가 끝났는지 체크하고 끝났으면 큐를 점검하여 다음 환자를 받아들인다. 큐에 대기 중인 환자가 없을 경우 프로세스를 종료한다. 환자가 해당 엔티티에 도착한 경우에는 환자의 타입에 따라 목적지가 현재 엔티티인지 체크하고 목적지가 현재 엔티티라면 자신의 큐에, 아니라면 환자의 다음 경로로 환자를 보낸다.



[그림 1] PIOS의 시뮬레이션 순서도

Entrance는 병원 입구를 표현하기 위한 엔티티의

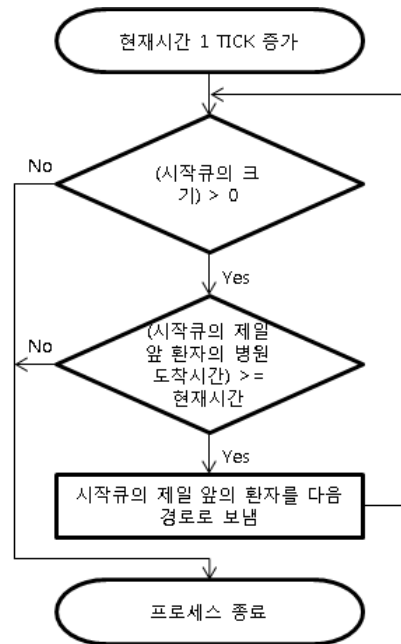


[그림 2] 엔티티의 수행 순서도

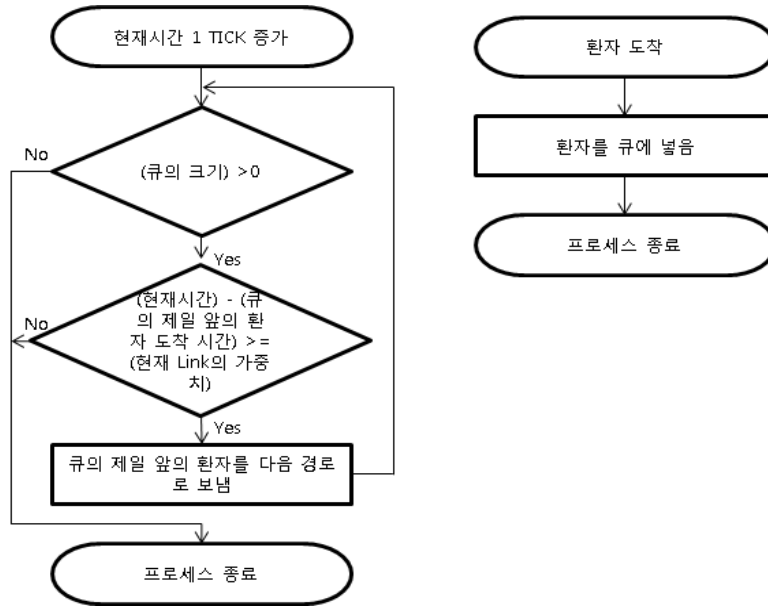
하위클래스로서 병원에 도착하는(시물레이션 시스템에 들어오는) 환자와 병원에서 진료를 마치고 떠나는(시물레이션 시스템을 떠나는) 환자를 각각의 큐에 넣는다. 시물레이션을 본격적으로 시작하기 전에 환자를 생성하기 때문에 환자 생성 이후 시물레이션이 진행되는 과정에서 Entrance는 시물레이션 시간이 1틱 증가할 때에만 가지고 있는 큐에서 제일 앞에 있는 환자가 도착한 시간과 시물레이션 시간을 비교하고 환자가 도착했을 경우 환자를 큐에서 꺼내어 다음 경로로 보낸다. [그림 3]은 Entrance에서 이루어지는 작업을 나타낸 순서도이다.

링크는 병원 내의 환자가 이동할 수 있는 경로(복도, 에스컬레이터 등)를 표현하기 위한 클래스로서 엔티티와 마찬가지로 시물레이션 시간이 증가했을 때나 환자가 도착했을 때 작업을 수행한다. 링크는 큐를 가질 필요가 없고 실제 경로에서도 큐가 존재하진 않지만 시물레이션 결과를 위해서 가상의 큐의 내부에 이동하는 환자들이 존재하는 것으로 모델링 하였다. [그림 4]에서와 같이 시물레이션 시간이 1틱 증가하면 시물레이터는 링크의 큐 내부에

환자가 존재하는지 보고 제일 먼저 도착한 환자가 링크를 벗어날 때가 되었는지 체크한다. 이때 링크



[그림 3] Entrance의 수행 순서도



[그림 4] 링크의 수행 순서도

의 가중치를 참고하는데 가중치는 환자가 해당 링크를 이동할 때 걸리는 시간을 의미한다. 해당 환자가 링크를 벗어날 때가 되었으면 환자를 다음 경로로 보내고 다시 큐에 있는 환자의 도착 시간을 살펴서 같은 작업을 해준다. 또 환자가 해당 링크에 도착하는 경우 링크의 큐에 넣고 대기시킨다.

도착시간의 정확성 등의 변수가 있을 수 있으나 현재의 PIOS에서는 시스템을 단순화하기 위하여 그러한 요소까지 반영하고 있지는 않다. 시간대에 따라 환자가 주어진 분포에 따라 발생하며 비율에 따라 예약, 비예약이 결정되게 된다. 각 엔티티에서 예약 환자는 비예약 환자보다 서비스 우선권을 갖는다고 가정한다.

2.2 PIOS 데이터 입력

본 장에서는 사용자가 입력해야 하는 PIOS 데이터에 대해서 살펴본다. PIOS에서 입력 받는 정보는 크게 환자 도착 정보, 엔티티 정보, 프로세스 정보 등 3가지이다.

환자도착 정보는 <표 2>의 예와 같이 시간대 별로 환자가 몇 명 도착하는지 입력 창을 통해 입력 받게 되며 예약 환자와 비예약 환자를 구분하여 입력 받을 수 있다. 환자의 도착 수 이외에도 환자가 도착하는 시간 간격의 분포를 선택할 수 있으며 환자는 시간대 별로 해당 분포를 따라 발생되어 Entrance의 시작 큐에 저장되어 있다가 차례대로 나온다. 현실문제에서는 예약환자의 경우 no-show,

<표 2> 환자 도착 정보

| 시작 시각 | 종료 시각 | 비예약 | 예약 | 시간 분포 |
|--------------|--------------|-----|----|-------------|
| 08 : 00 : 00 | 09 : 00 : 00 | 37 | 10 | Exponential |
| 09 : 00 : 00 | 10 : 00 : 00 | 39 | 10 | Exponential |
| 10 : 00 : 00 | 11 : 00 : 00 | 29 | 10 | Exponential |
| 11 : 00 : 00 | 12 : 30 : 00 | 33 | 15 | Exponential |
| 12 : 30 : 00 | 13 : 30 : 00 | 0 | 0 | Exponential |
| 13 : 30 : 00 | 14 : 00 : 00 | 54 | 0 | Exponential |
| 14 : 00 : 00 | 15 : 00 : 00 | 15 | 5 | Exponential |
| 15 : 00 : 00 | 16 : 00 : 00 | 7 | 10 | Exponential |
| 16 : 00 : 00 | 17 : 00 : 00 | 5 | 10 | Exponential |
| 17 : 00 : 00 | 18 : 00 : 00 | 2 | 0 | Exponential |

<표 3>에 예를 보이는 엔티티 정보는 시뮬레이션을 수행하고자 하는 병원에서 처리하는 프로세스들을 나타내는 정보로서 각 행은 서비스 프로세스를 의미한다. 프로세스를 위해 입력해야 하는 정보는 열에 표시되는데 시뮬레이션 경험이 없는 병원 사용자들을 고려하여 최대한 단순하게 입력할 수 있게 하였다. 첫째 열은 각 엔티티들의 이름을 나타낸다. 둘째 열은 프로세스의 타입을 나타내며 selection과 FIFO 중 한 가지 값을 갖게 된다. 여기서 selection은 같은 엔티티(예 : 의사 진료)를 두 번 이상 거쳐야 할 때 항상 첫 번째 방문한 엔티티에서 서비스를 받아야 한다는 것을 의미하며 FIFO는 첫 번째 방문한 엔티티와 상관없이 가장 빨리 서비스를 받을 수 있는 엔티티에서 서비스를 받는 것을 의미한다. 예를 들어, [그림 5]와 같이 환자 A가 진료실 1을 거친 후에 검사실 1을 방문하고 다시 진료실 1과 진료실 2중에서 어디로 갈지 정하는 부분이 있다고 가정했을 때 진료실의 프로세스 타입이 FIFO라면 환자 A는 진료실 1과 진료실 2중에서 빠르게 서비스를 받을 수 있는 곳에 갈 수가 있다. 그러나 프로세스 타입이 selection이라면 환자 A는 이미 이전에 진료실 1을 방문했었기 때문에 진료실 2가 서비스를 제공할 수 있고 기능이 같다고 하더라도 진료실 1을 방문해야만 하는 경우이다.

<표 3>의 3열은 그림번호로 엔티티를 표시하기 위해 미리 주어진 그림(icon)의 번호를 의미하고 4열 엔티티 수는 해당 엔티티의 자원개수를 의미한

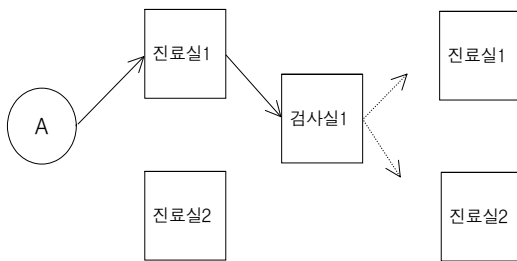
다. [그림 5]와 같이 동일한 기능을 하는 진료실이 2개 라면 엔티티 수는 2로 지정하면 된다. 이 엔티티 수에 따라서 5열부터 차례대로 4개의 열에 대해서 엔티티의 각 자원에 대한 서비스 분포 타입과 분포의 파라미터 값을 추가로 입력해야 한다. 예를 들어, 접수에 해당하는 엔티티에 자원이 2개 존재하고 지수 분포를 따를 경우 2개의 자원에 대해 모두 시간분포와 해당 파라미터를 입력해야 한다. 각 시간분포 타입에는 Exponential 이라고 입력할 수 있으며 파라미터 1에는 지수 분포의 평균값을 입력하고 지수분포는 파라미터를 하나만 사용하므로 파라미터 2에 입력된 값은 사용되지 않는다. 만약 시간 분포타입이 정규분포(Normal)라면 평균과 분산값을 파라미터로 입력한다. 모든 값은 PIOS의 입력 창을 통해 입력 받도록 되어 있다.

프로세스 정보는 환자가 거쳐야 하는 프로세스를 정의한 것이며 <표 4>와 같이 임의로 환자의 프로세스 타입을 나누고 환자의 예약 유무(예약, 비예약)에 따라서 거치는 프로세스를 입력한 것이다. <표 4>에서 첫째 열의 번호는 환자 타입을, 예약여부는 <표 2>에서 표시했었던 환자 타입의 예약 여부를, 비율은 예약 여부 조건에서 해당 타입이 차지하는 비율을 나타내며 그 이후의 열들은 환자 타입이 거쳐야 하는 엔티티들의 이름을 나타낸다. <표 4>의 예에서는 비예약 환자의 50%는 1번, 40%는 2번, 10%는 3번 프로세스를 거친다는 의미이다. 예약환자의 20%, 30%, 50%는 4번, 5번, 6번 프로세스를

<표 3> 엔티티 정보

| 엔티티 이름 | 프로세스 타입 | 그림 번호 | 엔티티 수 | 시간분포 | 파라미터 1 | 파라미터 2 | 그림 번호 | ... |
|-----------------|---------|-------|-------|-------------|--------|--------|-------|-----|
| AskDoctor | FIFO | 20 | 2 | Exponential | 104.1 | 0.0 | 20 | ... |
| BloodDraw | FIFO | 17 | 1 | Exponential | 80.2 | 0.0 | | ... |
| BloodPressure | FIFO | 13 | 1 | Exponential | 40.0 | 0.0 | | ... |
| CheckBreast | FIFO | 26 | 1 | Exponential | 583.3 | 0.0 | | ... |
| CheckEar | FIFO | 14 | 1 | Exponential | 14.5 | 0.0 | | ... |
| CheckLung | FIFO | 18 | 1 | Exponential | 218.3 | 0.0 | | ... |
| CheckSpecialEar | FIFO | 19 | 1 | Exponential | 374.8 | 0.0 | | ... |

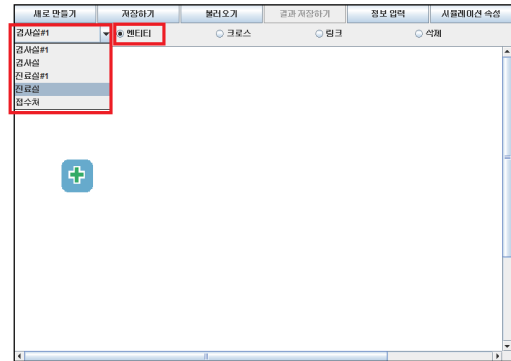
거친다. 이 정보들은 PIOS의 프로세스 입력 창을 통하여 입력된다. 한편 이러한 프로세스의 구성은 단순한 선형적인 흐름뿐만 아니라 반복 등의 다양한 흐름에 대해서도 표현이 가능하다. 예를 들어, 프로세스 3은 Reception → Consult → Reception → X-ray → Consult와 같이 서비스가 반복되는 복잡한 프로세스를 표현하고 있다. 그리고 분기와 같은 프로세스흐름도 프로세스를 추가로 정의하고 각 프로세스의 비율을 조정해 줌으로써 표현 가능하다.



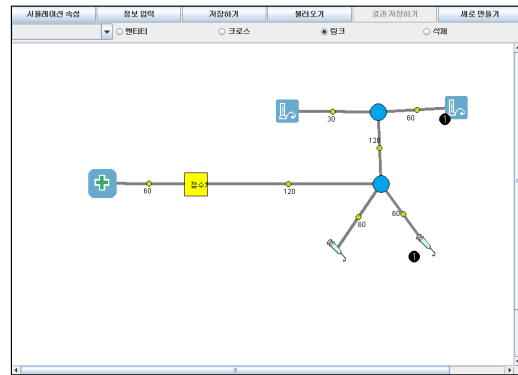
[그림 5] 프로세스 타입의 결정 예

2.3 레이아웃 구성

PIOS에서 시뮬레이션을 위한 레이아웃의 구성은 [그림 6]과 같이 단순한 마우스 클릭으로 완성할 수 있다. 엔티티 정보가 입력된 후에는 [그림 6](a)와 같이 시뮬레이션 화면상에 배치할 수 있도록 엔티티 리스트가 제공되는데, 리스트에 나타나는 순서대로 각 엔티티의 위치를 마우스로 클릭하면 된다. 배치된 엔티티는 마우스로 끌어다 놓기(drag and drop)로 위치 조정이 가능하다. 링크와 크로스도 같은 방법으로 [그림 6](b)와 같이 구성하면 된다.



(a) 엔티티 배치



(b) 크로스 및 링크 배치

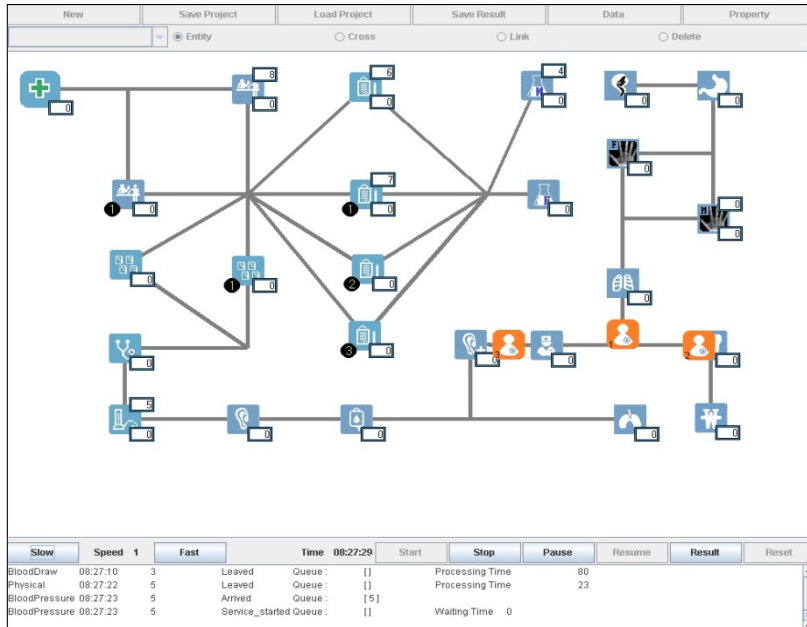
[그림 6] Layout 구성

2.4 PIOS 수행 결과

앞에서 기술한 것과 같이 정해진 데이터를 모두 입력하고 그에 맞게 병원의 레이아웃을 그려주면 [그림 7]과 같이 [Start]버튼을 클릭하여 시뮬레이션을 수행할 수 있다. 그림에서 선들로 연결된 사각형

<표 4> 프로세스 정보

| 번호 | 예약여부 | 비율 | 프로세스 1 | 프로세스 2 | 프로세스 3 | 프로세스 4 | 프로세스 5 | 프로세스 6 | ... |
|----|------|-----|-----------|---------------|---------------|---------------|---------------|----------|-----|
| 1 | No | 0.5 | Reception | FillUp | Order | Physical | BloodPressure | CheckEar | ... |
| 2 | No | 0.4 | Reception | FillUp | Physical | BloodPressure | CheckEar | | ... |
| 3 | No | 0.1 | Reception | Consult | Reception | X-ray | Consult | | ... |
| 4 | Yes | 0.2 | FillUp | Order | Physical | BloodPressure | CheckEar | | ... |
| 5 | Yes | 0.3 | FillUp | Physical | BloodPressure | CheckEar | FillUp | Physical | ... |
| 6 | Yes | 0.4 | Physical | BloodPressure | CheckEar | FillUp | | | |

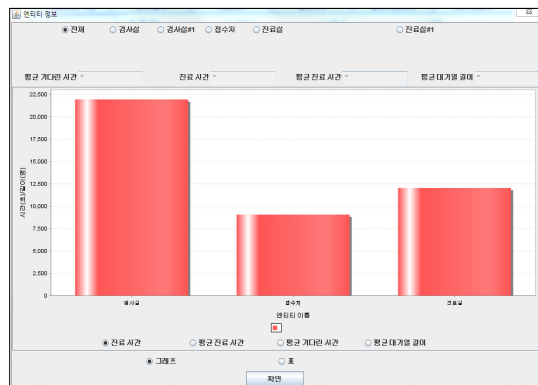
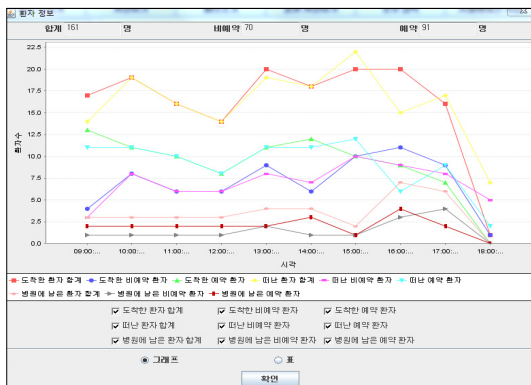


[그림 7] PIOS의 수행화면

들은 엔티티들을 나타내며 각 선들은 링크를 나타낸다. 엔티티의 왼쪽 아래의 숫자는 진료실 1이나 진료실 2와 같이 같은 엔티티지만 엔티티의 자원 개수가 여러 개인 경우 구분하기 위한 표시를 나타내며 각 엔티티의 우측 하단의 숫자는 현재 엔티티에서 서비스를 받기 위해서 대기하고 있는 환자의 숫자를 의미한다. 엔티티 우측 상단에 숫자가 있는 경우는 현재 환자에게 서비스를 제공하고 있다는 것

을 의미하며 숫자는 환자의 고유 ID를 나타낸다. 시뮬레이션의 속도는 화면 아래의 [Slow], [Fast] 버튼을 사용하여 조절할 수 있으며, 일시 정지하거나 (Pause) 정지하거나(Stop) 다시 수행(Resume)할 수 있다.

시뮬레이션이 종료되면 시뮬레이션을 수행한 결과를 확인할 수 있다. PIOS에서 제공하는 결과는 크게 환자 정보, 엔티티 정보, 프로세스 정보가 있다. 환



[그림 8] 시뮬레이션 환자 정보 결과 화면

자 정보의 경우 도착한 환자 수, 떠난 환자 수, 병원에 남은 환자 수를 시간대별로 정리하여 [그림 8]과 같이 보여준다. 엔티티 정보의 경우 각 엔티티 별로 평균 기다린 시간, 진료 시간, 평균 진료 시간, 평균 대기열 길이와 같은 정보들을 보여준다. 그리고 프로세스 정보의 경우 각 프로세스마다 환자들이 머무른 시간, 기다린 시간, 이동 시간, 진료 시간과 같은 정보들을 확인할 수 있다. 이러한 통계값들은 사용자가 그래프 형태나 표 형태를 선택하여 조회할 수 있으며 표 형태의 값들은 선택하여 Excel 등의 다른 소프트웨어로 복사하여 추가 분석 작업을 수행할 수도 있다.

3. PIOS의 유효성 검증

PIOS의 활용을 위해서는 PIOS를 구성하는 시뮬레이션 로직의 정합성 및 PIOS를 이용한 시뮬레이션 결과물의 유효성의 검증이 필요하다. 이를 위해 상용 시뮬레이션 도구인 AutoMod를 이용한 gene-

ric 시뮬레이션 모델의 실험결과와 PIOS를 이용하여 만든 시뮬레이션 모델의 결과를 비교하였다. AutoMod를 이용하여 만들어진 generic 시뮬레이션 모델의 유효성(validity)은 포항의 증견 병원들의 사례를 통하여 실제 병원의 운영을 충분히 시뮬레이션하고 있음을 확인받는 과정을 거쳤다. 본 연구에서는 PIOS의 유효성을 검증하기 위해 AutoMod를 이용하여 생성한 generic 모델과의 비교를 수행하였다. 이를 위해 <표 5>와 같이 환자도착 분포와 프로세스 시간 분포를 달리하는 세 가지의 시나리오를 이용하였다. 환자가 거쳐야 하는 엔티티 정보 및 순서는 건강검진의 예가 되는 프로세스(Reception → FillUp → Order → Physical → BloodPressure → CheckEar → Urinalysis → BloodDraw → AskDoctor → CheckTeeth → X_ray)로 PIOS와 Auto Mod 모델에서 동일하게 지정하였다. 테스트에 사용된 시나리오에서는 오전 9시부터 저녁 6시까지 약 200명의 환자가 도착한다.

세 가지 시나리오에 대해 PIOS의 결과와 Auto-

<표 5> 상용 시뮬레이션 프로그램(AutoMod)와 PIOS의 비교 환경

| 테스트 | 환자 도착 분포 | 프로세스 분포 | t-test 종류 | 시뮬레이션 시간 |
|-----|-------------|-------------|------------|----------|
| 1 | Exponential | Constant | Two sample | 30 Days |
| 2 | Constant | Exponential | Two sample | 30 Days |
| 3 | Exponential | Exponential | Two sample | 30 Days |

<표 6> 상용 시뮬레이션 프로그램(AutoMod)와 PIOS 모델의 평균 시간 비교를 위한 two-sample t-test의 p-value (유의 수준 = 0.05)

| 시나리오 | 모델 | 머문 시간 | | | 대기 시간 | | | 서비스 시간 | | | 도착 시간 | | |
|------|---------|--------|-------|---------|--------|-------|---------|--------|-------|---------|-------|-------|---------|
| | | 평균 | 표준 편차 | p-value | 평균 | 표준 편차 | p-value | 평균 | 표준 편차 | p-value | 평균 | 표준 편차 | p-value |
| 1 | PIOS | 2006.7 | 255.7 | 0.12 | 526.7 | 277.7 | 0.12 | 1156.0 | 0.0 | | 172.6 | 22.5 | 0.89 |
| | AutoMod | 2020.0 | 340.6 | | 498.7 | 315.6 | | 1156.0 | 0.0 | | 170.3 | 27.6 | |
| 2 | PIOS | 2672.9 | 344.3 | 0.84 | 1171.1 | 330.0 | 0.88 | 1151.7 | 29.9 | 0.46 | 180.0 | 0.0 | |
| | AutoMod | 2698.2 | 306.7 | | 1199.3 | 290.5 | | 1148.9 | 26.2 | | 180.0 | 0.0 | |
| 3 | PIOS | 3434.7 | 372.4 | 0.49 | 1928.3 | 368.4 | 0.50 | 1160.5 | 25.0 | 0.50 | 187.0 | 26.9 | 0.89 |
| | AutoMod | 3383.1 | 219.5 | | 1877.9 | 220.3 | | 1155.2 | 37.6 | | 186.8 | 23.4 | |

Mod 모델의 결과의 two sample t-test의 값은 <표 6>과 같다. 세 가지 시나리오 모두 5% 유의 수준에서 결과값이 통계적으로 같다는 귀무가설을 기각할 수 없었다. 즉, PIOS를 이용한 시뮬레이션 결과값이 상용 소프트웨어 AutoMod를 이용한 결과값과 비슷하다고 결론 지을 수 있으며 이는 PIOS의 유효성을 보이는 결과이다.

4. 결 론

본 연구에서는 외래환자 진료 프로세스 개선을 위한 generic 시뮬레이션 모델인 PIOS를 개발하고 그 유효성을 검증하였다. PIOS는 자바 애플릿을 기반으로 만들어졌으며 외래 프로세스 시뮬레이션을 별도의 프로그래밍 없이 데이터 입력 및 레이아웃 배치만으로 수행할 수 있도록 설계되었다. 따라서 시뮬레이션이 익숙하지 않은 병원 관계자들도 PIOS를 이용하여 시뮬레이션 모델을 쉽게 만들어보고 결과를 확인할 수 있다. 복잡한 프로그래밍을 요구하는 대부분의 시뮬레이션 도구들과는 달리 사용자는 시간대 별로 병원에 도착하는 환자수와 같은 환자 도착 정보, 진료 프로세스의 자원개수나 처리하는데 소요되는 시간과 같은 엔티티 정보, 병원에 도착하는 환자들이 거쳐야 하는 진료순서와 같은 프로세스 정보 등 세 가지 정보와 병원의 레이아웃만 입력한다면 시뮬레이션을 수행할 수 있다. PIOS의 유효성은 상용 시뮬레이션 도구인 AutoMod와의 결과 비교를 통해 검증하였다.

PIOS가 기존의 다른 병원 시뮬레이션 모델들과 구분되는 특징은 크게 두 가지가 있다. 첫째, 외래환자 프로세스용의 generic 시뮬레이터로서 별도의 프로그래밍 없이 데이터 입력 및 레이아웃 배치만으로 시뮬레이션 모델을 만들고 수행해 볼 수 있다. 둘째, 대부분의 상용시뮬레이션 도구와는 다르게 PIOS는 인터넷을 통해 무료로 이용할 수 있기 때문에 많은 병원들에서 별도의 소프트웨어를 설치하지 않고서도 쉽게 이용할 수 있다. 따라서 병원의 규모와 수익이나 전문적인 지식의 유무와 상관없이

쉽고 편리하게 외래환자 프로세스와 관련한 시뮬레이션을 수행할 수 있어 병원들의 프로세스 개선에 많은 도움을 줄 수 있을 것이다.

PIOS는 <http://logistics.postech.ac.kr/pios>에서 한글과 영문 버전으로 제공된다. 사이트 내에서 튜토리얼을 포함한 매뉴얼과 Q&A 게시판 등이 제공되고 있다. PIOS는 공개된 이후 보건 행정 담당자들의 관심을 받고 있으며 실제 병원들에 적용하는 사례들이 만들어지고 있다. PIOS는 실제 병원들에 적용되면서 추가적인 요구사항들을 반영하여 지속적으로 업그레이드 될 계획이다. 반복시뮬레이션 기능, 환자타입에 따른 엔티티에서의 소요시간 설정 기능, 엔티티에서 리소스 사용 기능 등이 곧 추가될 예정이다. PIOS가 잘 활용되어 국내 의료서비스 분야에 경영과학적 기법의 적용을 확산하는 작은 기폭제 역할을 할 것으로 기대한다.

본 연구에서 개발한 외래 프로세스 진단용 시뮬레이터는 수정, 보완을 거쳐서 다른 용도의 시뮬레이터로 발전시킬 수 있을 것이다. 최근에 포괄수가제의 도입으로 표준 임상경로(Clinical pathway: CP)에 대한 관심이 높아지고 있다. 표준 임상경로는 제조업에서의 표준 공정순서와 같은 개념으로 어떤 질환의 환자가 의료기관에서 받는 의료서비스의 순서 경로이다. CP 개발을 위해서는 각 질환에 대해서 어떤 임상경로가 의료기관과 환자의 입장에서 가장 효과적일 것인가에 대한 시나리오 분석이 필요하다. 시뮬레이션은 CP를 적용하였을 때 환자의 전체 진료 기간이 얼마나 소요될 지와 의료기관의 자원의 활용성 등의 영향도 분석에 효과적인 도구가 될 것이다. 본 연구에서 개발한 시뮬레이터를 보완, 확장하면 CP 분석용으로 활용될 수 있을 것이다. 또한 입원환자 프로세스 진단용 시뮬레이터도 필요하다. 입원환자 프로세스는 외래환자 프로세스와는 다른 요소들을 가지고 있다. 가령, 간호사들이 주기적으로 환자를 방문하여 환자의 상태를 체크해야 한다. 이런 요소들을 포함하여 적정 의료인력이 얼마가 되어야 하는 지 등에 대한 시뮬레이션을 할 수 있을 것이다.

참 고 문 헌

- [1] 구정인, 이종성, 김병인, “다양한 의료 프로세스를 고려한 generic 시뮬레이션 모델 개발”, 『대한산업공학회 추계학술대회』, 2009.
- [2] 권방현, “종합병원 외래환자의 접수/수납 대기시간의 최소화를 위한 시뮬레이션 분석”, 『한국QA학회 논문집』, 제2권, 제1호(1997), pp.97-108.
- [3] 대한민국정부, 『서비스 산업 경쟁력 강화 종합대책』, 2006.
- [4] 박하영, “의료서비스 산업과 산업공학 : 병원부문”, 『IE매거진』, 제9권, 제2호(2002).
- [5] 보건복지부, 『2011 보건복지통계연보』, 제57호(2011).
- [6] 위주영, 전여운, 김병인, “외래 환자 시스템을 위한 generic 시뮬레이션 모델 개발”, 『대한산업공학회 춘계공동학술대회』, 2008.
- [7] 장훈, 이태식, “병원 외래 환자 예약 방법 설계를 위한 시뮬레이션 최적화 기법의 효과 평가에 관한 연구”, 『대한산업공학회/한국경영과학회 춘계공동학술대회 논문집』, 2010.
- [8] 전희선, 박연식, 김승환, “외래환자 대기시간의 측정을 위한 시각적 시뮬레이션 모델”, 『한국통신학회 1997년 공개토론회』, 1997.
- [9] 정봉수, “의료서비스 산업 현황 및 환경변화”, 『KIS Credit Monitor』, 2009.
- [10] 최재혁, 이배진, 강창욱, 최경업, 김정미, “병원 약제부의 적정약사수를 결정하기 위해 대기시간을 고려한 시뮬레이션 모델 개발”, 『공업경영학회지』, 제21권(1998), pp.223-232.
- [11] Aytug, H. and C.A. Dogan, “A framework and a simulation generator for kanban-controlled manufacturing systems,” *Computers and Industrial Engineering*, Vol.34, No.2(1998), pp.337-350.
- [12] Harper, P.R. and H.M. Gamlin, “Reduced outpatient waiting times with improved appointment scheduling : a simulation modeling approach,” *OR Spectrum*, Vol.25(2003), pp.207-222.
- [13] Jun, J.B., S.H. Jacobson, and J.R. Swisher, “Application of discrete-event simulation in health care clinics : a survey,” *Journal of the Operational Research Society*, Vol.50(1999), pp.109-123.
- [14] Kim, B., S. Jeong, J. Shin, J. Koo, J. Chae, and S. Lee, “A layout-and data-driven generic simulation model for semiconductor fabs,” *IEEE Transactions on Semiconductor Manufacturing*, Vol.22, No.2(2010), pp.225-231.
- [15] Lee, S., H. Cho, and M. Jung, “A conceptual framework for the generation of simulation models from process plans and resource configuration,” *International Journal of Production Research*, Vol.38, No.4(2000), pp.811-828.
- [16] Lee, Y.T. and C.R. McLean, “A neutral data interface specification for simulating machine shop operations,” *Production Planning and Control*, Vol.17, No.2(2006), pp.143-154.
- [17] Lowery, J.C., “Design of hospital admissions scheduling system using simulation,” *the Proceedings of Winter Simulation Conference*, (1996), pp.1199-1204.
- [18] Mackulak, G.T., F.P. Lawrence, and T. Colvin, “Effective simulation model reuse : a case study for AMHS modeling,” *the Proceedings of Winter Simulation Conference*, (1998), pp. 979-984.
- [19] Mathewson, S.C., “The application of program generator software and its extensions to discrete event simulation modeling,” *IIE Transactions*, Vol.16, No.1(1984), pp.3-18.
- [20] Paul, R.J. and J. Kuljis, “A generic simulation package for organising outpatient clinic-

- ics," *the Proceedings of Winter Simulation Conference*, (1995), pp.1043-1047.
- [21] Pidd, M., "Guidelines for the design of data driven generic simulators for specified domains," *Simulation*, Vol.59, No.4(1992), pp.237-243.
- [22] Raikundalia, G.K., M. Mastan, and C.A. Bain, "A web-based visual simulator for hospital management using discrete event simulation," *Research Journal of Information Technology*, Vol.3, No.1(2011), pp.55-67.
- [23] Rohleder, T.R., D.P. Bischak, and L.B. Baskin, "Modeling patient service centers with simulation and system dynamics," *Health Care Management Science*, Vol.10(2007), pp. 1-12.
- [24] Son, Y.J., A.T. Jones, and R.A. Wysk, "Component based simulation modeling from neutral component libraries," *Computers and Industrial Engineering*, Vol.45(2003), pp.141-165.
- [25] Steele, M.J., M. Mollaghasemi, G. Rabadi, and G. Cates, "Generic simulation models of reusable launch vehicles," *the Proceedings of Winter Simulation Conference*, (2002), pp. 747-753.
- [26] Wijewickrama, A.K.A., "Simulation analysis for reducing queues in mixed-patients' outpatient department," *International Journal of Simulation Modelling*, Vol.5, No.2(2006), pp. 56-68.
- [27] Wy, J., S. Jeong, B. Kim, J. Park, J. Shin, H. Yoon, and S. Lee, "A data-driven generic simulation model for logistics embedded assembly manufacturing lines," *Computers and Industrial Engineering*, Vol.60, No.1(2011), pp.138-147.