Journal of Korea Multimedia Society Vol. 23, No. 5, May 2020(pp. 710-720)
https://doi.org/10.9717/kmms.2020.23.5.710

지진해일로 인한 해안 침수 분석을 위한 셀 오토마타 기반의 시뮬레이션 모델 개발: 광안리 해변 사례 연구

주재우†, 주준모†, 김동민†, 이동훈†, 최선한†

A Tsunami Simulation Model based on Cellular Automata for Analyzing Coastal Inundation: Case Study of Gwangalli Beach

Jae Woo Joo†, Jun Mo Joo†, Dong Min Kim†, Dong Hun Lee†, Seon Han Choi†

ABSTRACT

Tsunami occurred by a rapid change in the ocean floor is a natural disaster that causes serious damage worldwide. South Korea seems to be out of the range of this damage, but it is quite possible that South Korea will fall within the range due to the long-distance propagation features of tsunami and many earthquakes occurred in Japan. However, the analysis and preparation for tsunami have been still insufficient. In this paper, we propose a tsunami simulation model based on cellular automata for analyzing coastal inundation. The proposed model calculates the range of inundation in coastal areas by propagating the energy of tsunami using the interaction between neighboring cells. We define interaction rules and algorithms for the energy transfer and propose a software tool to effectively utilize the model. In addition, to verify and tune the simulation model, we used the actual tsunami data in 2010 at Dichato, Chile. As a case study, the proposed model was applied to analyze the coastal inundation according to tsunami height in Gwangali Beach, a famous site in Busan. It is expected that the simulation model can be a help to prepare an effective countermeasure against tsunami and be used for a virtual evacuating training.

Key words: Tsunami Simulation, Cellular Automata, Coastal Inundation

1. 서 론

해저에서 발생한 지진 또는 화산 활동 등의 급격한 지각변동은 그 위의 물을 이동시키게 되고, 이때 발생한 거대한 위치 에너지가 해일의 형태로 해안가에 도달하는 것을 지진해일(Tsunami) 이다[1]. 바람에 의한 해일과 달리 지진해일은 해저 깊은 곳에서부터 들어 올려진 수십억에서 수조 톤에 달하는 물의 위치 에너지에서 비롯된 운동에너지를 내재하고 있기 때문에 단 1 m의 파고라도 큰 파괴력을 가진다. 일례로 2004년 남아시아 대지진으로 인해 발생한 지진해일은 동남아시아 해안 전역에 큰 침수 피해와 함께 35만 명 이상의 사망자를 초래하였으며, 2011년 동일본 대지진으로 유발된 지진해일은 일본의 동쪽 해안에 원자력 발전소 사고를 포함한 천문학적인 피해를 초래하였다. 한국의 경우 3면이 바다로 둘러싸

※ Corresponding Author : Seon Han Choi, Address: (48513) 45 Yongso-ro, Nam-gu, Busan, Korea, TEL : +82-51-629-6240, FAX : +82-51-629-6230, E-mail : shchoi@pknu.ac.kr
Receipt date : Mar. 30, 2020, Revision date : Apr. 28, 2020
Approval date : May. 6, 2020
† Dept. of IT Convergence & Application Eng., Pukyong National University (E-mail : jaewoo@pukyong.ac.kr)
(E-mail : glps2167@pukyong.ac.kr)
(E-mail : dmk218@pukyong.ac.kr)
(E-mail : ldh9380@puykong.ac.kr)
※ This research was financially supported by the Ministry of Trade, Industry, and Energy (MOTIE), Korea, under the “Regional Innovation Cluster Development Program(P0004797)” supervised by the Korea Institute for Advancement of Technology (KIAT).

여 있지만 지진 발생 빈도와 규모가 상대적으로 작으며, 일본이 방파제 역할을 하여 해일의 전파를 다소 막아주기 때문에 지진해일에 대한 분석과 대비가 미진한 편이다. 하지만 1960년 칠레 대지진으로 인한 지진해일이 일본에 큰 피해를 입혔듯이, 동심원 형태로 전달되는 지진해일은 전달 거리가 굉장히 멀기 때문에, 비록 지진이 잘 발생하지 않더라도 한국이 지진해일로 피해를 입을 가능성을 무시할 수 없다. 또한, 일본에 자주 발생하는 지진은 한국에 지진해일로 인한 피해를 초래할 수 있다. 예를 들어서 1983년 일본의 아키타현 서쪽 앞바다에서 발생한 지진으로 인한 해일은 강원도 동해안에 인명 피해를 초래하였으며, 1993년 홋카이도 오쿠시리섬 북서 해역에서 발생한 지진해일은 울릉도에 큰 침수 피해를 입혔다.

지진해일이 자주 발생하는 경우 실제 사례와 데이터를 바탕으로 효과적인 분석과 대비책을 개발할 수 있지만, 한국과 같이 지진해일이 거의 발생하지 않는 경우 이러한 방법이 적절하지 않으며 또한 다른 나라의 사례를 그대로 적용하기에도 어려움이 있다. 이러한 경우에 시뮬레이션 기법은 효과적인 방법이 될 수 있다. 지금까지 지진해일에 대한 다양한 시뮬레이션 모델들이 제안된 바 있으며, 이들은 크게 두 가지로 분류할 수 있다. 첫 번째는 물리적인 요소들을 고려한 편미분 방정식을 수치 해석하여 유체의 흐름을 묘사하는 방식으로, Gato와 Ogawa[2]는 장파의 특성을 나타내는 지진해일의 전파특성에 착안하여 이를 수치 해석할 수 있는 모델을 제시하였으며, Mercado와 McCann[3]은 이를 사용하여 1918년 푸에르토리코 지진해일에 대한 발생, 전파 및 쳐오름(Runup)을 분석하였다. Imamura *et al.*[4]는 1992년도 인도네시아 플로레스(Flores) 섬의 지진해일을 분석하기 위해 산사태로 인한 파동 발생 모델인 원형 아크 슬립(Circular-arc slip model)을 적용하였다. Titov와 Gonzalez[5]는 지진해일의 사전 감지와 예측을 위하여 지진해일의 진행 단계를 나누어서 수치 방정식을 통해 묘사하는 MOST (Method of splitting tsunami) 모델을 제시하였다[6]. Koh *et al.*[7]는 SWE (Shallow water equations)을 사용하여 발생원으로부터 연안 지역으로의 지진해일에 대한 심해 전파를 시뮬레이션하였다.

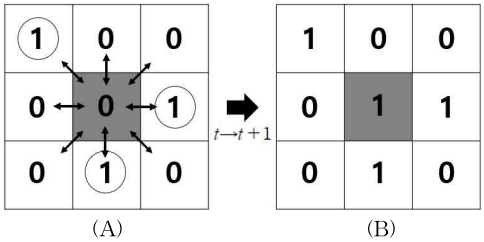

Fig. 1. An example of celullar automata: Conway's game of life (0: death, 1: live).

두 번째 방식은 셀 오토마타(Cellular automata)를 바탕으로 에너지의 확산을 모의하는 방식이다. 셀 오토마타는 잘게 쪼개어진 각 셀이 사전에 정의된 규칙을 바탕으로 이웃 셀들과의 상호작용을 통해 셀의 상태를 변경하므로 거시적인 흐름을 효과적으로 분석할 수 있는 이산형 시뮬레이션 기법으로 컴퓨터 보안, 도시 공학, 수학, 물리학, 생물학, 재난학 등의 다양한 분야에서 활용 된다[8-12]. Fig. 1은 2차원 셀 오토마타의 예로, 널리 알려진 콘웨이의 라이프 게임(Conway's Game of Life)을 나타낸다. Mohamed와 Rajasekaran[13]은 2차원 셀 오토마타를 활용하여 다양한 해양 지형 조건에서 지진해일의 확산을 모의하였으며, 이들은 지진해일의 복잡한 움직임을 보다 정밀하게 확인하기 위해 기존 모델을 이웃 셀들과의 상호작용이 증가하는 육각형 셀 오토마타 모델로 확장하였다[14]. 지진해일의 경우 첫 번째 방법에 비해서 셀 오토마타를 활용하는 두 번째 방법에 대한 연구는 상대적으로 덜 이루어졌다. 수치 해석 방법은 셀 오토마타에 비해서 보다 정확하게 모의할 수 있으나 고려해야 할 파라미터가 많으며 넓은 영역을 분석하기 위해서는 굉장히 많은 계산량을 필요로 한다. 반면에 셀 오토마타 모델은 수치 해석 방법에 비해서 세부적인 정확도가 떨어질 수 있지만 보다 넓은 영역의 확산과 움직임을 보다 효율적으로 확인할 수 있으며, 모델을 간략화할 수 있다는 장점이 있다.

본 논문에서는 셀 오토마타 기반의 지진해일 시뮬레이션 모델을 제안한다. 기존의 모델들이 대부분 지진해일의 생성과 전파를 분석하기 위해서 개발된 반면에, 본 모델은 지진해일로 인한 피해를 사전 분석하고 이를 최소화하기 위한 효과적인 대비책을 개발하기 위하여 연안 지역의 침수범위를 확인하는 것에 그 목적을 둔다. 제안하는 모델은 각 셀이 가지고 있는 에너지 전파 규칙에 따라서 연안 지역에서 지진해일 에너지가 확산되는 것을 시뮬레이션한다. 본 논문에서는 모델을 효과적으로 활용하기 위해서 실제 지

도 이미지에서 셀을 추출하는 도구를 포함하는 소프트웨어를 개발하였으며, 2010년의 칠레 디차토(Dichato) 지역의 지진해일로 인한 연한 침수 데이터를 바탕으로 개발한 모델을 튜닝하고 검증하였다. 그리고 이를 사용하여 부산의 광안리 지역에 대해서 해일의 높이에 따른 침수범위를 분석하였다. 본 시뮬레이션 모델을 바탕으로 한국의 지진해일에 대한 효과적인 대비책을 개발하고, 또한 가상 대피 훈련용으로도 활용될 수 있을 것이라 기대한다.

본 논문의 구성은 다음과 같다. 2장에서 셀 오토마타 기반의 지진해일 시뮬레이션 모델을 제시하며, 시뮬레이션 모델에 대한 가정 및 에너지 전달 규칙과 알고리즘을 소개한다. 또한, 모델을 효과적으로 사용하기 위한 소프트웨어 도구와 실제 데이터를 바탕으로 모델의 신뢰도 향상을 위한 검증을 수행한다. 3장에서는 개발된 모델을 바탕으로 광안리 지역에 대한 해일 높이에 따른 침수 범위 분석을 수행하고, 4장에서 결론을 맺는다.

2. 지진해일 시뮬레이션 모델 개발

2.1 시뮬레이션 가정사항

본 연구에서는 모델의 가정사항을 다음과 같이 정하였다. 첫 번째로, 해일은 방향성을 가지며 이 시뮬레이션의 목표는 해안에서의 침수범위를 구하는 것이기 때문에 초기방향을 해안으로 설정하였다. 두 번째로, 해수욕장은 일반적으로 경사가 완만하므로 고도를 고려하지 않고 시뮬레이션 모델을 구현하였다. 마지막으로, 모델을 구현함에 있어서 조금 더 쉽게 다루기 위해 해일은 건물을 파괴할 수 없고 건물 내부로 흐르는 물은 고려하지 않기로 하였다. 또 유체의 마찰력으로 인한 에너지 감소량은 Blasius′ equation[15]에 근거하여 10^{-10} 보다 작은 값을 가지므로, 시뮬레이션 결과에 미치는 영향이 매우 적을 것으로 예상하여 고려하지 않았다.

2.2 셀 오토마타 기반의 시뮬레이션 모델 개발

본 연구에서는 2차원 셀 오토마타를 사용하는데, 인접한 8방향에 위치한 셀과 상호작용하는 무어 이웃(Moore neighborhood)과 인접한 상하좌우의 4방향에 위치한 셀과 상호작용 하는 폰 노이만 이웃(Von Neumann neighborhood)이 있다[16]. 4방향의 셀과의 상호작용으로 8방향의 상호작용을 나타낼 수 있다. 예를 들어 우측상단 방향의 경우에, 우측과의 상호작용 이후 상단으로의 상호작용을 통해 표현할 수 있고, 이로 인해 본 모델을 간략화할 수 있으므로 폰 노이만 이웃을 바탕으로 모델을 개발하였다. 좌측하단을 (0, 0)이라고 할 때, C는 좌표(x, y)의 한 셀을 나타내며, Q_c는 셀 C와 상호 작용하는 셀의 집합이다. 4방향으로 상호작용하는 폰 노이만 이웃 셀 F, L, R, B는 식 (1)과 같이 정의된다. 하지만 해일의 진행 방향을 묘사하기 위해 각 셀이 가진 고유의 진행 방향에 따라 F, L, R, B는 각기 다른 셀로 정의된다. 시각 t일 때, 셀 C에서의 진행 방향 d_C^t는 식 (2)와 같이 정의되며, Fig. 2와 같이 +y 방향을 1로 설정하고 이를 기준으로 시계 방향으로 2는 +x 방향, 3은 - y 방향, 4는\ -x 방향을 의미한다. 식 (3)은 이 진행 방향에 따라서 각기 다르게 정의되는 Q_c을 나타낸다.

$$Q_c = \{F, L, R, B\} \tag{1}$$

$$d_C^t \in \{1, 2, 3, 4\} \tag{2}$$

$$d_C^t \begin{cases} 1: F=(x,y+1), L=(x+1,y), R=(x-1,y), B=(x,y-1) \\ 2: F=(x+1,y), L=(x,y+1), R=(x,y-1), B=(x-1,y) \\ 3: F=(x,y-1), L=(x-1,y), R=(x+1,y), B=(x,y+1) \\ 4: F=(x-1,y), L=(x,y-1), R=(x,y+1), B=(x+1,y) \end{cases} \tag{3}$$

하나의 셀은 주변 셀들과 에너지를 교환하며 상호작용을 한다. 시각 t일 때, 셀 C에서 셀 F, L, R로 전달할 수 있는 에너지를 E_C^t라 하며, 이 에너지의 단위는 J($N \cdot m$)이다. 셀 C는 셀 F, L, R, B로부터 에너지를 전달받는다. 전달받은 에너지의 크기와 방향에 따라 다음 진행 방향 d_C^{t+1}가 결정이 되며, 전달받은 에너지의 합 E_C^{t+1}은 다음 시각 $t+1$에 d_C^{t+1}을 바탕으로 정의된 셀 F, L, R에 전달된다. E_{stop}은 다른 셀과 상호작용할 수 있는 최소 에너지로 $E_C^t > E_{stop}$을 만족하면, 셀 C에서 셀 F, L, R로 에너지를 전달할 수 있다. 에너지 상호작용은 셀의 진행 방향과 주변 셀의 상태에 따라 4가지 경우로 분류할 수 있다.

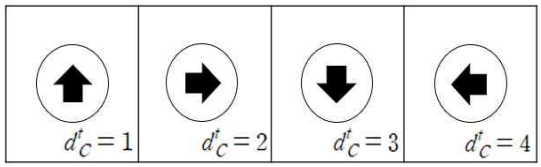

Fig. 2. The four directions of cell C.

1) 진행 방향의 셀과 인접한 양옆의 셀이 건물이 아닐 경우

이 경우 셀 C는 셀 F, L, R로 모두 에너지를 전달할 수 있다. 시각 t일 때, $E^t_{C\to F}$는 현재 위치에서 진행 방향, $E^t_{C\to L}$는 진행 방향의 왼쪽, $E^t_{C\to R}$는 진행 방향의 오른쪽으로 전달하는 에너지이며, 각각 식 (4) 및 (5)와 같이 정의된다. λ_P는 각각의 셀마다 가지고 있는 고유의 에너지 감쇄율이며, ϵ은 에너지를 전달 가능한 진행 방향으로 분배하는 비율이다. 셀 고유의 에너지 감쇄율인 λ_P는 $0 < \lambda_P \le \delta$의 범위를 가지며, 여기서 δ는 건물에 부딪혔을 때 에너지 감쇄율로, 다음 경우에서 보다 자세히 설명할 것이다. 침수된 셀의 경우 앞에서 정의한 가정사항에 따라 유체에서의 마찰력을 고려하지 않으므로 0의 λ_P값을 가진다. ϵ의 값은 모든 셀에서 동일한 값을 가지며, 앞서 분류한 4가지 경우에 따라서 다르게 활용된다. 하지만 분배 비율의 총합은 항상 1이다. 예를 들어서 본 경우에 진행 방향으로 ϵ만큼의 에너지를 전달하고, 양옆으로는 각각 $\frac{1-\epsilon}{2}$만큼의 에너지를 전달한다.

$$E^t_{C\to F} = E^t_C \times \epsilon \times \lambda_P \tag{4}$$

$$E^t_{C\to L} = E^t_{C\to R} = E^t_C \times \frac{1-\epsilon}{2} \times \lambda_P \tag{5}$$

반면에, 셀 C는 셀 F, L, R, B로부터 모두 에너지를 전달받을 수 있으며, 시각 t일 때, $E^t_{F\to C}$는 현재 진행 방향, $E^t_{L\to C}$는 진행 방향의 왼쪽, $E^t_{R\to C}$는 진행 방향의 오른쪽, $E^t_{B\to C}$는 진행 방향의 뒤쪽으로부터 전달받는 에너지를 의미한다. 하지만 셀의 다음 시각 $t+1$의 에너지 E^{t+1}_C는 이들 전달받은 에너지의 합으로만 정의되지 않는다. 왜냐하면, 셀 C가 전달하는 에너지와 전달받는 에너지의 간의 상쇄가 발생하기 때문이다. 예를 들어서 셀 F에 전달하는 에너지는 $E^t_{C\to F}$이고 셀 F로부터 전달받은 에너지는 $E^t_{F\to C}$이므로, 셀 F로부터 전달받은 실질적인 에너지는 $E^t_{F\to C} - E^t_{C\to F}$로 계산될 수 있다. 하지만 전달받는 에너지보다 전달하는 에너지의 크기가 더 큰 경우, 즉 $E^t_{F\to C} - E^t_{C\to F}$ 값이 음수일 경우 실질적으로 에너지를 받지 않는다. 이를 고려하여 셀 F로부터 전달받는 에너지는 $\max(0, E^t_{C\to F} - E^t_{F\to C})$와 같이 정의된다. 셀 L과 R에 대해서도 이와 동일하게 정의되며, 단 셀 B의 경우 에너지를 전달하지 않으므로 B로부터 받은 에너지가 그대로 C의 다음 시각 에너지에 더해진다. 결과적으로 셀 F, L, R, B로부터 전달받은 실질적인 에너지의 합 E^{t+1}_C은 식 (6)과 같이 정의되며, Fig. 3은 이를 설명하기 위한 예를 나타낸다. d^t_C = 2일 때의 에너지의 흐름을 나타낸다. 화살표의 길이는 에너지의 크기를 나타내고, 얇은 화살표는 현재 셀과 이웃 셀의 에너지 흐름, 굵은 화살표는 실질적인 에너지의 흐름을 나타낸다. 셀 F, R, B의 경우 셀 C 방향으로 실질적인 에너지 전달이 일어나는 반면에, 셀 L의 경우 셀 C에서 셀 L로 전달된다. 즉 셀 L로부터 받은 에너지보다 셀 C에서 셀 L로 전달하는 에너지가 더 크다는 것을 의미한다. 따라서 앞서 설명한 식 (6)에 의하여, E^{t+1}_C는 셀 F, R, B에서 셀 C를 가리키는 굵은 화살표의 합이다.

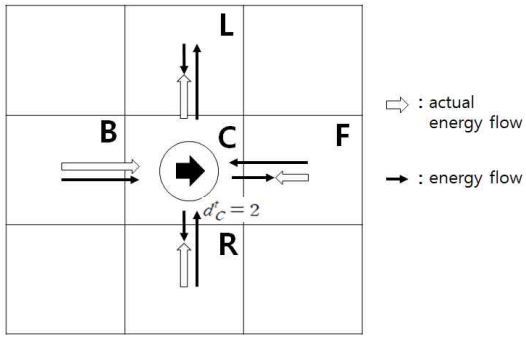

Fig. 3. According to (3), the four neighborhood cells: F, R, L, B are determined by d^t_C = 2. The thin arrows indicate the energy flow between cells, and the thick arrows mean the actual energy flow between cells.

$$E^{t+1}_C = \max(0, E^t_{F\to C} - E^t_{C\to F}) + \max(0, E^t_{L\to C} - E^t_{C\to L}) + \max(0, E^t_{R\to C} - E^t_{C\to R}) + E^t_{B\to C} \tag{6}$$

셀 C의 다음 진행 방향 d^{t+1}_C는 현재 셀의 방향을 결정하는 에너지 기준 D_C와 시각 t에서 전달받은 에너지 중 가장 큰 에너지와의 비교를 통해 결정되며, 알고리즘 1을 이를 상세하게 나타낸다. 먼저 셀 C가 전달받은 에너지 중 가장 큰 에너지를 전달한 셀 P를 구한다. 셀 P로부터 셀 C가 받은 실질 에너지 $E^t_{P\to C} - E_{C\to P}$와 D_C를 비교하여, D_C가 더 클 경우 현재 진행 방향을 그대로 유지하며 (즉, $d^{t+1}_C = d^t_C$), 반대

로 $E^t_{P\to C}-E_{C\to P}$가 더 클 경우 진행 방향은 셀 P에서 셀 C의 방향으로 결정된다. 하지만 셀 P는 d^t_C에 따라 상대적으로 결정되는 값이므로, 절대 방향인 d^{t+1}_C을 계산하기 위해서는 d^t_C을 고려하여 이를 치환해야 한다. 예를 들어서, d^t_C가 1이고 셀 P가 셀 L인 경우, 셀 L에서 셀 C의 방향은 2이며, d^t_C가 3일 경우 4가 된다. 알고리즘 1은 이를 치환하는 방법을 나타낸다. P의 값에 따라서 x값이 정해지고, 현재의 방향에 x의 값을 더한 후 4로 나눈 나머지에 1을 더한 값이 d^{t+1}_C이 된다. d^t_C에 x를 더했을 때 범위가 0～6까지고 방향은 4가지뿐이므로, 4로 나눈 나머지의 값을 사용한다. 4로 나눈 나머지 값의 범위는 0～3이고 논문 초기에서 정의한 방향의 범위는 1～4이기 때문에 1을 더하여 방향을 설정한다. 방향이 바뀌는 경우 D_C는 $E^t_{P\to C}-E_{C\to P}$로 변경된다.

2) 진행 방향의 셀이 건물일 경우

이 경우 셀 C는 셀 F가 건물이기 때문에, 셀 L, R로만 에너지를 전달할 수 있으며, 셀 C가 셀 L, R로 전달하는 에너지는 식 (7)과 같이 정의된다. δ는 건물 셀마다 가지고 있는 고유의 에너지 감쇄율이며 모든 건물 셀에서 동일하다. 일반적으로 유체가 땅에 흐르는 경우보다, 장애물에 부딪혔을 때의 에너지 감소량이 많으므로, δ는 $\lambda_P<\delta\le 1$의 범위를 가진다. 건물 셀의 경우 앞의 가정사항에서 말한 바와 같이 건물이 파괴될 수 없으므로, δ의 값은 해일이 건물에 부딪힐 때마다 적용된다. 반면에 에너지를 전달받을 경우, 셀 C는 F, L, R, B로부터 모두 에너지를 전달받을 수 있으므로 경우 1)과 같이 E^{t+1}_C은 식 (6)에 의해 결정된다. 단, 셀 F는 건물이기 때문에 에너지가 0이다.

$$E^t_{C\to L}=E^t_{C\to R}=E^t_C\times\delta\times\frac{1-\epsilon}{2}\times\lambda_P \tag{7}$$

셀 C의 다음 진행 방향 d^{t+1}_C는 기존의 진행 방향의 셀이 건물이기 때문에 기존의 진행 방향으로 진행할 수 없다. 따라서 진행 방향을 변경해야 하며, 알고리즘 2는 이를 상세하게 나타낸다. 먼저 셀 F에서 d^t_C에 대해 상대적으로 정해지는 좌측과 우측을 *Find_L*, *Find_R*로 탐색한다. *Find_L*, *Find_R* 함수는 각각 좌측과 우측으로 탐색하여 가장 가까운 길까지의 거리를 반환하는 함수이다. 좌측과 우측의 길까지의 거리가 가까운 쪽으로 d^{t+1}_C을 설정한다. 예를 들어 *Find_L* > *Find_R*일 경우, 우측으로의 길이 더 가까우므로, d^{t+1}_C을 d^t_C 기준 우측으로 설정한다. 반대로, *Find_L* < *Find_R*일 경우, d^{t+1}_C을 d^t_C 기준 좌측으로 설정한다. 만약 좌측과 우측의 거리가 같을 경우, 0과 1중 무작위 값을 반환하는 random(1)을 사용하여 그 값에 따라 0이면 좌측, 1이면 우측으로 다음 진행 방향을 설정한다. Fig. 4는 이를 설명하기 위한 예를 나타낸다. 셀 F가 건물일 때, 에너지 전달 규칙 및 방향의 변화를 나타낸다. 시각 t에서 셀 F가 건물이므로, 셀 C에서 셀 F로 에너지를 전달하지 못하고

Algorithm 1. Set the next direction

```
Dir_Base(D_C, E^t_{F→C}, E^t_{R→C}, E^t_{L→C}, E^t_{B→C})
P ← argmax_{C'∈F,B,L,R} E^t_{C'→C} − E_{C→C'}
if E^t_{P→C} − E_{C→P} > D_C then
    D_C ← E^t_{P→C} − E^t_{C→P}
    switch P
        case 'F': x ← 1
        case 'R': x ← 2
        case 'B': x ← −1
        case 'L': x ← 0
    d^{t+1}_C ← (d^t_C + x) %4 +1
else then
    d^{t+1}_C ← d^t_C
```

Algorithm 2. Set the next direction in front of buildings

```
Dir_Building(d^t_C)
L ← Find_L(d^t_C, F), R ← Find_R(d^t_C, F)
if L > R then
    d^{t+1}_C ← d^t_C %4 + 1
else if R > L then
    d^{t+1}_C ← (d^t_C + 2)%4 + 1
else then
k ← random(1)
  if k = 0
      d^{t+1}_C ← (d^t_C + 2)%4 + 1
  else if k = 1
      d^{t+1}_C ← d^t_C %4 + 1
```

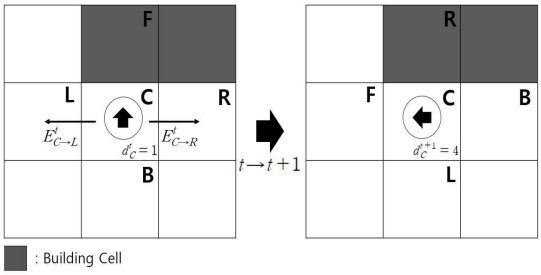

Fig. 4. An example of determining the next direction of cell C in front of buildings.

식 (7)에 의해 셀 L, R로 에너지를 전달한다. 그 후 알고리즘 2를 통해 d^{t+1}_C을 설정한다.

3) 진행 방향의 좌, 우중 하나의 셀이 건물인 경우

이 경우 셀 C는 셀 R이 건물일 경우 셀 F, L로 에너지를 전달하고, 반대로 셀 L이 건물일 경우 셀 F, R로 에너지를 전달할 수 있다. 셀 F로 전달하는 에너지는 1번 경우의 식 (4)를 사용하며, 셀 L 또는 R로 전달하는 에너지는 식 (8)과 같이 정의된다. Fig. 5는 이를 설명하기 위한 예를 나타낸다. 셀 R이 건물이므로, 셀 C에서 셀 F, L로만 에너지를 전달한다. 전달받은 에너지의 경우 셀 F, L, R, B로부터 모두 에너지를 전달받을 수 있으므로 셀 C는 경우 1)과 같이 E^{t+1}_C은 식 (6)에 의해 결정된다. 앞서와 마찬가지로 건물 셀의 에너지는 0으로 처리된다. 또한, 셀 C의 d^{t+1}_C도 마찬가지로 경우 1)과 같이 알고리즘 1에 의해 결정된다.

$$E^t_{C\to L\,\mathrm{or}\,R} = E^t_C \times (1-\epsilon) \times \lambda_P \tag{8}$$

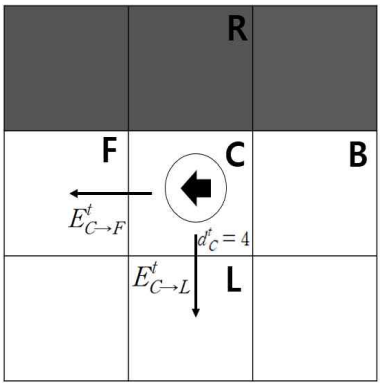

Fig. 5. If the right side of cell C is a building, it transfers its energy to the cells F and L.

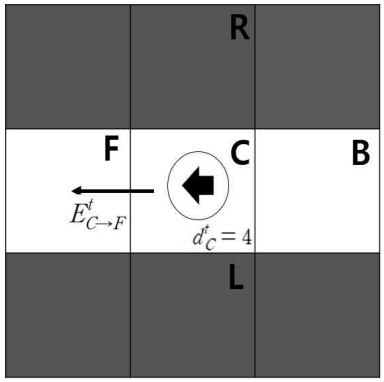

Fig. 6. If both sides of cell C are buildings, it transfers its energy only to the cell F.

4) 진행 방향 양쪽에 모두의 셀이 건물인 경우

이 경우 셀 C는 셀 L, R이 건물이기 때문에, 셀 F로만 에너지를 전달한다. 셀 C에서 셀 F로 전달하는 에너지는 식 (9)과 같이 정의된다. Fig. 6은 이를 설명하기 위한 예를 나타낸다. 셀 L, R이 건물이므로, 셀 C에서 셀 F로만 에너지를 전달한다. 경우 2)와 같이 건물 셀이 가진 에너지는 0이므로, 셀 C는 경우 1)과 같이 E^{t+1}_C은 식 (6)에 의해 결정된다. 셀 C에서 d^{t+1}_C도 마찬가지로 경우 1)과 같이 알고리즘 1에 의해 결정된다.

$$E^t_{C\to F} = E^t_C \times \lambda_P \tag{9}$$

알고리즘 3은 앞서 설명한 4가지 경우를 종합한 셀 오토마타 기반의 지진해일 시뮬레이션 모델인 TsunamiSim을 나타낸다. 먼저 해일 셀에 초기 에너지 E_{init}을 설정하고 모든 셀의 에너지가 E_{stop} 미만이 될 때, 즉 $\max(E^{tc}) < E_{stop}$ 조건을 만족할 때 시뮬레이션이 종료된다. 처음 반복문에서 각 셀에 대해서 주변 셀로 전달하는 에너지를 계산한다. 각 셀의 에너지 E^t_C가 E_{stop}보다 클 때 전달된다. $E^t_C > E_{stop}$을 만족했을 때, 에너지를 전달한다. 두 번째 반복문에서는 전달받은 에너지로 각 셀의 d^{t+1}_C과 E^{t+1}_C을 계산한다. set_case(C)는 앞서 설명한 4가지 경우 중 셀 C에서 적용되는 경우를 판단한다. Fig. 7은 알고리즘 3을 설명하기 위한 예를 나타낸다. 밝은 셀은 길(해일이 진행할 수 있는 셀)을 나타내며, 어두운 셀은 건물을 나타낸다. 화살표는 진행 중인 해일의 방향을 나타내며, 빗금 친 셀은 $E^t_C > E_{stop}$조건을 만족하지

Algorithm 3. TsunamiSim based on cellular automata

TsunamiSim(E_{init})

$t \leftarrow 0$
$E^t_{(0,0)} = E^t_{(0,1)} = \ldots = E^t_{(0,width)} \leftarrow E_{init}$
while $\max(E^{t\,C}) >= E_{stop}$
  foreach $\forall C$
    if $E^t_C > E_{stop}$ then
      switch set_case(C)
        case '1':
          $E^t_{C\to F} \leftarrow E^t_C \times \epsilon \times \lambda_P$
          $E^t_{C\to L} = E^t_{C\to R} \leftarrow E^t_C \times \frac{1-\epsilon}{2} \times \lambda_P$
        case '2':
          $E^t_{C\to L} = E^t_{C\to R} \leftarrow E^t_C \times \delta \times \frac{1-\epsilon}{2} \times \lambda_P$
        case '3':
          $E^t_{C\to F} \leftarrow E^t_C \times \epsilon \times \lambda_P$
          $E^t_{C\to L \text{ or } R} \leftarrow E^t_C \times (1-\epsilon) \times \lambda_P$
        case '4':
          $E^t_{C\to F} \leftarrow E^t_C \times \epsilon \times \lambda_P$
  end
  foreach $\forall C$
    switch set_case(C)
      case '1', '3', '4':
        $Dir_{Base}(D_C,\ E^t_{F\to C},\ E^t_{R\to C},\ E^t_{L\to C},\ E^t_{B\to C})$
      case '2':
        $Dir_{Building}(d^{t\,C})$
    $E^{t+1}_C \leftarrow \max(0, E^t_{C\to F} - E^t_{F\to C}) + \max(0, E^t_{C\to R} - E^t_{R\to C}) + \max(0, E^t_{C\to L} - E^t_{L\to C}) + E^t_{B\to C}$
  end
  $t \leftarrow t+1$
end

못해서 에너지를 전달할 수 없는 셀이다. 시각 t일 때부터 $t+2$일 때까지 시간순으로 나타낸다.

2.3 TsunamiSim 소프트웨어 개발

본 논문에서는 앞서 제안한 셀 오토마타 기반의 TsunamiSim을 효율적으로 활용하기 위하여 지도 변환기(Map translator, MapT)를 포함한 소프트웨어를 개발하였다. MapT는 시뮬레이션할 목표 지역의 지도 이미지를 TsunamiSim이 인식할 수 있는 셀로 바꾸어 준다. 잘 알려진 이미지 처리 라이브러리인 OpenCV를 활용하여 입력받은 지도 이미지 각 픽셀의 RGB 값을 분석하고, 그 값에 따라 셀의 상태를 판단하여 지도 데이터(Map data)를 만든다. TsunamiSim은 JAVA 프로그래밍 언어 기반으로 개발되었으며, MapT를 통해 전달받은 지도 데이터와 해일의 초기 에너지를 바탕으로 시뮬레이션을 진행하여 침수범위를 도출한다.

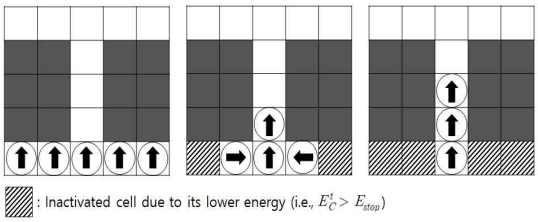

Fig. 7. An example of the energy flows based on TsunamiSim.

2.4 시뮬레이션 모델 검증: 칠레 디차토(Dichato) 지역

제안한 TsunamiSim의 시뮬레이션 결과는 모델 내부의 3가지 파라미터 ϵ, λ_P, δ에 영향을 받는다. 진

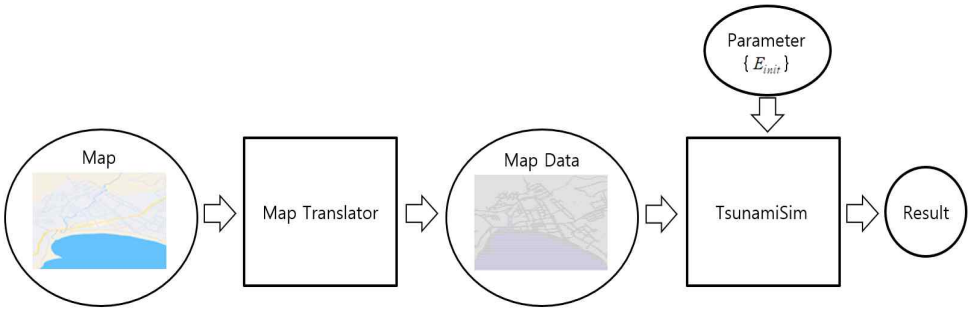

Fig. 8. Implementation of TsunamiSim, including the map translator.

행 방향으로 에너지를 분배하는 비율인 ϵ는 값이 커질수록 세로로 긴 모양으로 침수가 되며, 작을수록 가로로 긴 모양으로 침수된다. 셀마다 가지는 고유의 에너지 감쇄율 λ_p와 건물에 부딪혔을 때의 에너지 감쇄율인 δ는 값이 커질수록 에너지 감소가 커져서 침수범위가 줄어든다. 본 장에서는 칠레의 디차토 지역에서 2010년도에 발생한 지진해일을 바탕으로 위 3개 파라미터에 대한 적절한 값을 산출하였다. 먼저 디차토 지역의 지도를 사용하여 Map Translator을 통해 이에 대한 Map Data를 산출하였다. 그리고 디차토 지역의 해일 높이인 10 m에 근거하여 해일의 운동에너지와 위치에너지를 고려하여 초기 에너지 E_{init}을 4.05×10^{10}J로 설정하였다[17]. 3가지 파라미터의 특성을 고려하여 4가지 조합을 도출하였으며, Table 1은 이를 나타낸다. Table 1의 각 조합의 값은 Fig. 9에서 각 파라미터의 값의 변화에 따른 차이를 시각적으로 잘 보여주기 위해 편차를 의도적으로 크게 하였다. TsunamiSim 내부에는 실세계의 불확실성을 반영하기 위한 랜덤 변수가 포함되어 있으므로, 각 조합에 대한 반복 시뮬레이션을 통하여 확률적 노이즈를 줄인 정확한 침수범위를 도출하였다. 그리고 이를 디차토 지역의 2010년 해일에 의한 실제 침수범위와 비교하였다.

Fig. 9는 4가지 조합에 대한 침수범위를 디차토 지역의 지도 데이터 위에 나타내어 실제 침수범위와 비교한 결과이다. 굵은 선은 디차토 지역의 실제 침수범위를 나타내며, 나머지 선은 각 조합 P1～P4의 시뮬레이션 결과로 도출된 침수범위이다. 시뮬레이션 결과를 수치로 비교하기 위해서 실제 침수범위와 시뮬레이션 결과 간의 평균제곱근오차(Root mean square error, RMSE)를 계산하였으며, Table 2는 이를 나타낸다. Fig. 9와 Table 2의 결과를 바탕으로, P1이 실제와 가장 유사한 결과를 가지는 것을 확인할 수 있었으며, 이를 바탕으로 TsunamiSim의 파라미터를 P1의 조합으로 설정하였다.

Table 1. Parameter values of the four configurations

parameters / configurations	ϵ	λ_P	δ
P1	0.7	0.2	0.3
P2	0.1	0.7	0.9
P3	0.3	0.95	0.99
P4	0.5	0.99	0.999

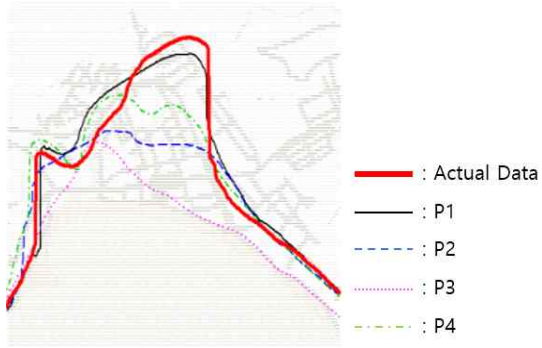

Fig. 9. Simulation results of the four configurations. The configuration P1 shows the similar results with the actual data.

Table 2. RMSE values of the four configurations

Configurations	RMSE
P1	25.273
P2	59.626
P3	95.544
P4	46.139

3. 부산 광안리 해안에 대한 사례 연구

광안리 해수욕장은 대한민국 부산광역시에 위치한 여름철 인기 피서지이다. 매년 800만 명 이상이 방문하며, 피서철 하루 관광객이 20～30만 명 정도에 이른다. 아직까지 광안리에 발생한 해일 사례에 대한 기록은 없지만, 일본에서 발생한 지진에 의해 우리나라 동해안의 여러 도시에 해일이 관측된 기록이 수차례 있어, 완전하게 해일로부터 안전하다고 할 수 없다. 따라서 본 사례 연구에서는 광안리 지역에 대해서 제안한 TsunamiSim을 바탕으로 해일의 높이에 따른 침수범위를 분석하고자 한다.

시뮬레이션을 수행하기 앞서서, 개발한 Tsunami Sim 소프트웨어의 MapT를 사용하여 시뮬레이션을 위한 광안리 지역의 지도 데이터를 추출하였으며. Fig. 10은 이를 나타낸다. 이 데이터에 대해서 해일의 높이를 3 m, 5 m, 10 m로 변경하며 광안리 지역의 침수범위를 도출하였다. Fig. 11은 10 m 높이에 대하여 시간에 따른 셀의 에너지 확산을 나타내며, Fig. 12는 시뮬레이션이 종료되었을 때 해일의 높이에 따른 최종 침수범위를 나타낸다. 해일의 높이가 높아질수록 더 많은 범위가 침수된 것을 확인할 수 있다. 앞서 언급한바 현재까지 광안리 해수욕장에 해일에 대한 기록이 없었기 때문에, 지진해일 대피소에 대한 인지도가 전무하며, 대피소와 대피 수칙에 대한 효과도 검증 등이 이루어진 적이 없었다. 본 시뮬레이션

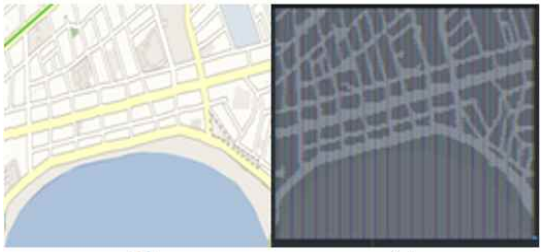

Fig. 10. Gwangalli beach: (a) real map image from Naver map[18] and (b) map data processed by MapT.

Fig. 11. Simulation progress when the height is 10 m.

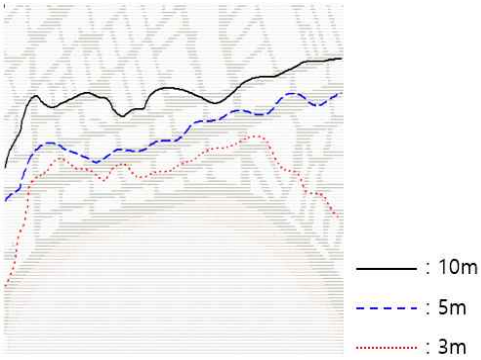

Fig. 12. Simulation results of costal inundation according to the height in Gwangalli.

결과는 이를 검증하고 개선하기 위한 근거 자료로 활용될 수 있으며, 더 나아가 지진해일에 대한 가상 대피 훈련 용도로도 활용될 수 있을 것이다.

4. 결 론

본 논문에서는 셀 오토마타를 바탕으로 지진해일로 인한 연안의 침수범위를 분석하는 시뮬레이션 모델을 개발하였다. 각 셀은 에너지와 진행 방향을 가지고 있으며 진행 방향에 따른 이웃 셀에 에너지를 전달하며. 이웃 셀들로부터 전달받은 에너지를 바탕으로 다음 에너지와 진행 방향을 결정한다. 에너지 교환이 불가능한 건물 셀의 위치에 따라서 에너지 상호작용을 4가지 경우로 구분하였고, 각 경우에 대하여 에너지 전달 및 진행 방향 결정 알고리즘을 제시하였다. 개발한 시뮬레이션 모델을 효과적으로 활용하기 위해서 실제 지도 이미지로부터 셀 데이터를 생성하는 지도 생성기를 포함한 소프트웨어를 개발하였으며, 모델의 신뢰도를 높이기 위해 칠레의 디차토 지역의 실제 지진해일 침수범위 데이터를 바탕으로 검증을 수행하였다. 사례 연구에서 부산의 광안리 지역에 대해서 지진해일의 높이에 따른 침수범위를 분석하였으며, 이 결과는 최적의 대피 장소를 설정하고 대피 매뉴얼을 검증하며, 또한 가상 지진해일 대피 훈련의 용도 등으로 효과적으로 활용될 수 있다. 향후 연구 계획은 본 논문에서 가정한 사항을 완화하여 제안된 모델을 보다 고도화하고, 또한 더 많은 실제 지진해일 침수 데이터를 바탕으로 모델을 검증하여 보다 정확한 결과를 도출할 수 있도록 한다. 또한, 본 모델과 연동하여서 대피 장소를 탐색하고 가상의

훈련을 진행할 수 있는 기능성 게임(Serious game)을 개발한다.

REFERENCE

[1] E. Bryant, *Tsunami: The Underrated Hazard*, Springer Praxis Books, New Zealand, 2008.

[2] UNESCO, *IUGG/IOC Time Project, IOC Manuals and Guides* No. 35, 1997.

[3] A. Mercado and W. McCann, "Numerical Simulation of the 1918 Puerto Rico Tsunami," *Natural Hazards,* Vol. 18, No. 1, pp. 57-76, 1998.

[4] F. Imamura, E. Gica, T. Takahashi, and N. Shuto, "Numerical Simulation of the 1992 Flores Tsunami: Interpretation of Tsunami Phenomena in Northeastern Flores Island and Damage at Babi Island," *Pure and Applied Geophysics,* Vol. 144, No. 3/4, pp. 555-568, 1995.

[5] V.V. Titov and F.I. Gonzalez, "Implementation and Testing of the Method of Splitting Tsunami(MOST) Model," *National Oceanic and Atmospheric Administration Technical Memorandum,* Vol. ERL PMEL-112, pp. 1-11, 1997.

[6] NOAA Pacific Marine Environmental Laboratory, https://nctr.pmel.noaa.gov/model.html (accessed September 15, 2019).

[7] H.L. Koh, S.Y. The, P.L. Liu, A.I. Ismail, and H.L. Lee, "Simulation of Andaman 2004 Tsunami for Assessing Impact on Malaysia," *Journal of Asian Earth Sciences*, Vol. 36, No. 1, pp. 74-83, 2009.

[8] T. Toffoli and N. Margolus, *Cellular Automata Machines: A New Environment for Modeling*, MIT Press, Massachusetts, 1987.

[9] U. Choi, S. Cho, and T. Kim, "Image Encryption Based on One Dimensional Nonlinear Group Cellular Automata," *Journal of Korea Multimedia Society*, Vol. 18, No. 12, pp. 1462-1467, 2015.

[10] J. Lee, H. Jang, and K. Rhee, "A Stream Cipher Using a Cellular Automata," *Journal of Korea Multimedia Society*, Vol. 5, No. 2, pp. 191-197, 2002.

[11] X. Li, X. Li, Y. Xio, and B. Jia, "Modeling Mechanical Restriction Differences Between Car and Heavy Truck in Two-lane Cellular Automata Traffic Flow Model," *Physica A*, Vol. 451, pp. 49-60, 2016.

[12] P.P. Chaudihuri, S. Ghosh, A. Dutta, and S.P. Dhoudhury, *A New Kind of Computational Biology: Cellular Automata Based Models for Genomics and Proteomics*, Springer Praxis Books, New Zealand, 2018.

[13] E.S. Mohamed and S. Rajasekaran, "Tsunami Wave Simulation Models Based on Hexagonal Cellular Automata," *World Scientific and Engineering Academy and Society Transactions on Fluid Mechanics*, Vol. 8, No. 3, pp. 91-101, 2013.

[14] E.S. Mohamed and S. Rajasekaran, "Performance Enhancement for Tsunami Wave Simulation Using Hexagonal Cellular Automata," *International Journal of Computer Applications*, Vol. 75, No. 9, pp. 36-43, 2013.

[15] Y. Nakayama and R.F. Boucher, *Introduction to Fluid Mechanics*, Elsevier, Netherlands, 1998.

[16] E. Jang, *The Science of Complexity*, Chonpascience, Seoul, 1999.

[17] S. Rahman Md, *Coastal Community Resilience to Tsunami: A Study on Planning Capacity and Social Capacity, Dichato, Chile*, Master's Thesis of Universidad Austal de Chile of Science, 2011.

[18] Naver Map, https://map.naver.com/v5/?c=14373381.2274296,4184589.3036289,15,0,0,0,dh (accessed December 12, 2019).

주 재 우

2015년~현재 부경대학교 IT융합응용공학과 학사과정
2019년~현재 부경대학교 시스템 모델링 시뮬레이션 연구실 연구원
관심분야 : 시스템 모델링 시뮬레이션, 셀 오토마타

주 준 모

2015년~현재 부경대학교 IT융합응용공학과 학사과정
2019년~현재 부경대학교 시스템 모델링 시뮬레이션 연구실 연구원
관심분야 : 시스템 모델링 시뮬레이션, 셀 오토마타

김 동 민

2015년~현재 부경대학교 IT융합응용공학과 학사과정
2019년~현재 부경대학교 인공지능 연구실 연구원
관심분야 : 인공지능, 시스템 모델링 시뮬레이션

이 동 훈

2015년~현재 부경대학교 IT융합응용공학과 학사과정
2019년~현재 부경대학교 시스템 모델링 시뮬레이션 연구실 연구원
관심분야 : DEVS, 에이전트 기반 시뮬레이션

최 선 한

2012년 한국과학기술원(KAIST) 전기및전자공학과 학사
2014년 한국과학기술원(KAIST) 전기및전자공학과 석사
2018년 한국과학기술원(KAIST) 전기및전자공학부 박사
2018년 한국생산기술연구원(KITECH) 선임연구원
2019년~현재 부경대학교 IT융합응용공학과 조교수
관심분야 : 시스템 모델링 시뮬레이션, 인공지능, 최적화, 데이터 모델링