

GPU를 이용한 확산모형 분석 도구: SNUDM-G

이 다 정^{1)†} 이 효 선²⁾ 고 성 룡¹⁾²⁾

¹⁾서울대학교 심리학과 ²⁾서울대학교 인지과학협동과정

이 논문에서는 계산 속도를 개선한 확산모형 분석 도구 SNUDM-G를 소개한다. 확산모형은 다양한 인지 과제를 설명하는 데에 적용되어 왔음에도 불구하고 계산적 어려움으로 인해 사용에 제한이 있었다. 특히 확산모형 분석 도구 중 하나인 SNUDM(고성룡 등, 2020)은 확산과정을 근사할 때 2만 개의 자료를 순차적으로 생성하기 때문에 처리 속도 면에서 단점이 있다. 이러한 한계를 극복하기 위해 확산과정을 무작위걷기 방법으로 근사하는 과정에서 그래픽처리장치(GPU)를 사용할 것을 제안한다. 그래픽처리장치를 사용하면 2만 개의 자료를 병렬로 생성할 수 있기 때문에 순차처리로 자료를 생성하는 것에 비해 분석의 속도를 높일 수 있다. GPU를 사용한 SNUDM-G와 CPU를 사용한 SNUDM으로 Ratcliff 등 (2004)의 실험 1 자료를 분석하고 매개변수 복구를 한 결과 SNUDM-G가 SNUDM보다 특정 매개변수에서 다소 높은 값을 추정하였으나, 계산 속도 면에서는 큰 차이로 SNUDM-G가 SNUDM보다 더 빠르게 매개변수를 추정하였다. 이 결과는 이 도구를 이용하여 다양한 인지 과제에 대해 보다 효율적인 확산모형 분석이 가능할 것임을 보여주며, 더 나아가 앞으로 그래픽처리장치를 이용하여 다양한 인지 모형의 처리 속도를 개선할 수 있음을 시사한다.

주요어 : 확산모형, GPU, 무작위걷기

† 교신저자: 이다정, 서울대학교 심리학과, (08826) 서울시 관악구 관악로 1 16동 M404호
연구 분야: 언어심리학, E-mail: ldj0716@snu.ac.kr

확산모형(Diffusion model)(Ratcliff, 1978)은 양자택일 과제의 의사결정 과정을 설명하는 인지 모형이다. 이 모형은 실험의 결과로 얻은 반응시간의 분포와 정답률을 모두 설명할 수 있다는 점에서 특징적이다. 이러한 특징을 바탕으로 확산모형은 다양한 인지 과제의 결과를 설명하는 데에 성공적으로 적용되어 왔다(Ratcliff, 2014; Ratcliff & McKoon, 2008; Ratcliff et al., 2016; Ratcliff et al., 1999; Ratcliff & Rounder, 1998; Voss et al., 2004; Arnod et al., 2015; Lerche & Voss, 2017). 그럼에도 불구하고 확산모형이 가지는 수학적, 계산적 어려움은 연구자로 하여금 확산과정을 근사하고 매개변수를 추정하는 서로 다른 방법을 제안하도록 만들었다. Brown 등 (2006)은 통계 미분방정식(stochastic differential equation)을 이용하여 확산과정을 근사하였으며, Voss와 Voss (2007)는 확산과정을 나타낼 수 있는 편미분방정식(Voss & Voss, 2008)을 사용하여 반응시간의 누적분포함수를 예측하고 Kolmogorov-Smirnov 방법(Voss et al., 2004)으로 매개변수를 추정하였다. Verdonck 등 (2016)은 Euler-Maruyama 방법을 사용하였으며, 그래픽처리장치(GPU)를 이용하여 처리의 속도를 높였다.

이 글에서는 확산과정을 근사하는 방법 중 하나인 무작위걷기(random walk) 방법(Ratcliff, 1978)을 사용하여 확산모형을 분석할 수 있는 도구를 제안하되, 무작위걷기 방법의 단점 중 하나인 계산 속도의 문제를 개선한 SNUDM-G를 제안하려 한다. 무작위걷기 방법은 시뮬레이션으로 다량의 자료를 생성하는 과정을 포함하기 때문에 다른 방법에 비해 계산 속도 면에서 한계가 있다. 이를 극복하기 위해 그래픽처리장치를 사용하여 무작위걷기 방법을 이용한 확산모형 분석도구를 만들고 기존의 무작위걷기 방법을 이용한 확산모형 분석도구 SNUDM과 비교하여 이 도구가 인지 과제를 분석하기에 더 유용한지 확인해보고자 한다.

확산모형

확산모형(Ratcliff, 1978; Ratcliff & McKoon, 2008)은 양자택일의 의사결정 과정이 연속적으로 정보를 누적하는 과정이라고 본다. 시작위치 z 에서 정보가 평균 v 의 속도로 위 또는 아래로 움직이면서 정보가 누적되다가 위쪽 경계나 아래쪽 경계(a 또는 0)에 도달하면 멈춘다. 그리고 이 정보 누적의 과정에 노이즈(noise)가 있다고 가정하며, 이 노이즈를 시행 내 변산도(within-trial variability)라고 한다. 확산모형에서 노이즈는 일반적으로 정규분포를 따른다고 가정하며, 조정 매개변수(scaling parameter)이다. 따라서 분석 시 추정하지 않고 0.1이나 1 등 고정된 값을 사용한다.

이 모형에서 반응시간은 의사결정 시간(decision time)과 비결정시간(nondecision time)의 합으로 나타낸다. 비결정시간은 정보를 부호화 하는 시간과 반응을 수행하는 시간의 합을 의미하는데, 이 둘을 구분하지 않고 하나의 매개변수 T_{er} 로 표현한다. 의사결정 시간은 시작위

치 z 부터 평균 v 의 속도로 정보가 누적되어 2개의 경계 a 와 0 중 하나를 처음으로 통과 할 때까지 걸린 시간(first passage time)이 된다. 그리고 각각의 경계가 선택된 반응의 결과를 나타낸다. 예를 들어 단어 재인 과제에서는 ‘예’와 ‘아니오’의 2가지 반응이 있고 이 중 긍정 반응을 a , 부정 반응을 0으로 표상한다면, ‘예’를 선택하는 것은 시작위치에서 정보가 누적되어 처음 도달한 경계가 a 인 것으로 표현할 수 있다.

확산모형의 매개변수는 의사결정 과정의 특정한 인지적 요인과 대응된다. 먼저 정보가 누적되는 평균 속도 v 즉 정보의 표집률(drift rate)은 값이 클수록 더 빠르고 정확한 반응을 만들어낸다. 다시 말해 표집률이 크면 반응시간과 오반응률이 모두 줄어들며, 표집률이 작으면 반응시간이 늘어나고 정반응과 오반응의 비율이 비슷해진다. 이 매개변수는 수행하는 과제의 난이도에 따라 달라진다고 알려져 있다(Ratcliff, 1978; Ratcliff & McKoon, 2008; Voss et al., 2004). 쉬운 과제는 높은 표집률과 관련이 있으며 반대로 어려운 과제는 낮은 표집률과 관련이 있다. 또한 이 매개변수는 개인차를 반영한다고 알려져 있는데, 예를 들어 Ratcliff 등 (2011)은 IQ가 표집률에 영향을 미친다는 사실을 발견하였다. 경계(boundary separation)를 나타내는 매개변수 a 는 클수록 반응시간이 길어지며 오반응이 줄어든다. 반대로 a 가 작아지면 반응시간이 줄어들고 노이즈의 영향이 커지게 되어 오반응이 증가하게 된다. 이 매개변수는 속도와 정확성에 대한 연구자의 실험 지시와 관련이 있다고 알려져 있다. 즉 속도의 강조는 경계의 하강과 관련이 있고 정확성의 강조는 경계의 상승과 관련이 있다(Ratcliff & Rouder, 1998). 또한 경계 매개변수는 나이와 관련이 있는데, 연령이 높은 참여자가 연령이 낮은 참여자보다 더 큰 경계값을 가진다는 것이 밝혀졌다(Starns & Ratcliff, 2010, 2012; Ratcliff et al., 2001). 시작위치를 나타내는 매개변수 z 는 2개의 경계 a 나 0에 가까워질수록 노이즈의 영향으로 인해 가까워진 경계의 반응률이 상승하고, 반응시간은 감소한다. 반대로 떨어진 경계의 반응률은 감소하고 반응시간은 증가하게 된다. 이 매개변수는 반응 편향과 관련이 있다고 알려져 있다(Voss et al., 2004). 참여자가 특정 반응을 기대할수록 시작위치 z 는 해당 반응의 경계에 가까워지게 된다. 특정 반응을 기대하지 않은 참여자는 위쪽 경계 a 와 아래쪽 경계 0의 중앙에서 정보의 누적이 시작된다.

시작위치 z , 경계 a , 표집률 v , 비결정시간 T_{er} 이외에 확산모형은 매개변수의 시행 간 변산도(across-trial variability)를 가정하고 있다(Ratcliff & Rouder, 1998; Ratcliff & Tuerlinckx, 2002). 이 가정은 정반응과 오반응의 속도 차이를 설명하기 위해 필요하다(Ratcliff et al., 1999; Ratcliff & Rouder, 1998; Ratcliff, 1981). 표집률은 정규분포를 따른다고 가정되며, 표집률 v 의 정확한 의미는 그 정규분포의 평균이다. 그리고 해당 분포의 표준편차를 η 라 한다. 시작위치 z 는 범위가 s_z 인 균등분포를 따른다고 가정된다. 마찬가지로 비결정시간 T_{er} 도 범위가 s_t 인 균등분포를 따른다고 본다. 이에 따라 확산모형의 매개변수는

시작위치 z , 경계 a , 표집률 v , 비결정시간 T_{er} 그리고 시작위치 z 의 변산도 매개변수인 s_z , 표집률 v 의 변산도 매개변수인 η , 비결정시간 T_{er} 의 변산도 매개변수인 s_t 등 7개로 구성된다.

GPU를 이용한 불연속 무작위건기 시뮬레이션

모형이 예측하는 정반응과 오반응의 반응시간 분포를 관찰된 반응시간 분포와 비교하기 위해 확산과정을 다양한 방법으로 근사할 수 있다. SNUDM-G에서는 연속적인 확산과정을 불연속적인 무작위건기로 근사하는 방법을 사용한다. Feller (1968)는 무작위건기에서 걸음의 크기(step size)가 작아지고 걸음의 수가 커지며 한 걸음이 특정 경계로 향할 확률이 0.5에 근접하면 확산과정을 무작위건기로 근사할 수 있음을 보였다. 즉 걸음의 크기 δ 와 한 걸음에 걸리는 시간 h 가 0에 수렴하고 걸음이 아래로 향할 확률 q 가 1/2에 수렴할 때, $(p - q)\delta/h$ 와 $4pq\delta^2$ 이 각각 특정 상수 v 와 s^2 에 수렴하게 된다. Ratcliff와 Tuerlinckx (2002)는 걸음의 크기와 한 걸음이 아래쪽 경계로 향할 확률을 한 걸음에 걸리는 시간과 노이즈 s 로 조정하여 각각 $\delta = s\sqrt{h}$, $q = 0.5(1 - v\sqrt{h/s})$ 로 정의하였다. 이에 따라 h 가 0에 수렴하면 δ 역시 0에 수렴하고 q 는 1/2에 수렴하게 되어, 시간 h 동안 움직인 위치의 평균 $(p - q)\delta/h$ 와 분산 $4pq\delta^2$ 은 위의 정리에 따라 각각 표집률의 평균 v 와 분산 s^2 이 된다. 마찬가지로 이 글의 모형에서도 걸음의 크기 δ 를 $s\sqrt{h}$ 로 정의하였고, 한 걸음이 위쪽 경계로 향할 확률 p 를 $0.5(1 + v\sqrt{h/s})$ 로 정의하여 사용하였다. 노이즈 s 는 0.1로 고정하여 사용하였으며(Ratcliff, 1978; Ratcliff & Tuerlinckx, 2002), 한 걸음에 걸리는 시간 h 는 0.5ms로 하였다. 따라서 SNUDM-G에서 확산과정은 시작위치 z 부터 단위 시간 h 동안 p 또는 $(1 - p)$ 의 확률로 위 또는 아래를 선택하여 δ 만큼 움직이며 이를 경계 a 나 0에 다다를 때까지 반복하는 것으로 시뮬레이션 된다. a 또는 0에 도착하는 것을 한 번으로 보고 이것을 2만 번 시행하여 반응률과 반응시간의 분포를 만들어낸다. 그리고 이때 만들어진 정반응의 반응시간 분포와 오반응의 반응시간 분포를 관찰된 반응시간 분포와 비교한다.

CPU를 이용하여 위에서 설명한 확산과정을 2만 번 수행하는 것은 상당한 시간을 필요로 한다. CPU에서는 시작위치에서 한 경계에 도착할 때까지의 무작위건기 과정 2만 번을 순차적으로 처리하기 때문이다. 이를 극복하기 위한 한 방법으로 그래픽처리장치의 병렬처리를 적용해볼 수 있다. 그래픽처리장치는 다량의 계산을 병렬처리로 수행할 수 있으며, 이를 C 프로그래밍 언어를 확장한 NVIDIA의 CUDA를 이용하여 사용할 수 있다. Verdonck 등 (2016)

은 그래픽처리장치를 이용하여 다양한 반응시간 모형을 Euler-Maruyama 방법으로 시뮬레이션했을 경우가 CPU를 이용했을 경우보다 효율적임을 보였다. 반응시간 모형뿐만 아니라 AI 탐색에서도 몬테카를로 무작위걷기(Monte Carlo Random Walk)의 계산 효율을 높이기 위해 그래픽처리장치를 사용하고 있다. Kuroiwa와 Fukunaga (2018)는 그래픽처리장치를 사용한 배치 몬테카를로 무작위걷기(batch Monte Carlo Random Walk)가 CPU에서의 그것보다 효율적임을 보였다. 이 글의 모형에서는 확산과정을 불연속 무작위걷기 과정으로 근사하고 반응시간 분포를 얻기 위해 2만 번의 무작위걷기 과정을 반복하되, 이에 소요되는 시간을 줄이기 위해 그래픽처리장치를 사용하여 2만 번의 무작위걷기 과정을 병렬처리 하도록 한다.

시뮬레이션으로 자료를 생성하는 부분 이외의 분석 과정은 CPU를 이용하여 계산한다. 모형이 예측한 반응시간 분포는 분위수를 이용한 카이제곱 방법으로 관찰된 반응시간 분포와 비교된다. 이를 위해 먼저 관찰된 정반응과 오반응의 반응시간 분포에서 각각 0.1, 0.3, 0.5, 0.7, 0.9에 해당하는 분위수를 구하고, 모형이 예측한 반응시간 분포를 앞서 구한 분위수를 이용하여 각각 6개의 구간으로 나누는 후 각 구간에 포함된 반응의 비율을 계산한다. 그리고 이 비율과 관찰된 반응시간 분포의 비율(0.1, 0.2, 0.2, 0.2, 0.2, 0.1) 간의 차이를 구한다. 두 비율의 차는 네이만(Neyman) 카이제곱으로 계산한다. 네이만 카이제곱은 일반적인 카이제곱과 다르게 $\Sigma(O - E)^2 / O$ (O : 관찰값, E : 기댓값)로 계산된다. 즉 나누는 값이 기댓값이 아닌 관찰값이다. 관찰값과 기댓값의 차이를 구하는 방법에는 여러 가지가 있지만, 네이만 카이제곱의 값을 최소화하는 것은 다항 우도를 최대화 하는 것과 같기 때문에 네이만 카이제곱 방법을 사용하는 것은 분위 우도 방법을 사용하는 것과 같은 결과를 얻을 수 있다(Rarcliff & Teurlinckx, 2002). 또 이 방법을 쓰면 기댓값이 0이 되어 일반적인 카이제곱을 계산할 수 없는 문제를 예방할 수 있다. 즉 관찰 분포의 분위수로 예측된 분포의 분위수를 나누었을 때, 특정 구간의 빈도가 0이 될 수 있는데 이때 일반적인 카이제곱은 분모를 기댓값으로 두기 때문에 계산할 수 없는 문제가 발생한다. 네이만 카이제곱은 이러한 문제로부터 자유롭기 때문에 SNUDM-G에서는 이 방법을 사용한다.

매개변수를 최적화 하는 방법으로는 SIMPLEX(Nelder & Mead, 1965; Luersen et al., 2003)를 사용하며, 초깃값으로는 EZ-diffusion(Wagenmakers et al., 2007)을 사용한다. EZ-diffusion으로 표집률 ν 와 경계 a 그리고 비결정시간 T_{er} 의 초깃값을 구하고 a 의 절반으로 시작위치 z 의 초깃값을 정한다. 변산도를 나타내는 매개변수 η , s_z , s_t 는 0부터 0.005의 범위에서 무작위로 정해지도록 한다. 모든 매개변수는 그 의미를 생각할 때 0보다 작아질 수 없기 때문에 0 이상의 값을 가지도록 제약을 둔다. SIMPLEX를 구현한 C 언어 프로그램으로는 <http://www.mikehutt.com>에서 공개된 소스를 찾아 사용하였다. SNUDM-G는 <http://psych.snu.ac.kr/snudm/>에서 내려 받아 사용할 수 있다.

SNUDM-G의 사용법과 적용

Nvidia Titan Xp를 써서 구현한 SNUDM-G의 사용법을 설명하고 이 모형을 자료에 적용해 보려고 한다. Ratcliff 등 (2004)의 실험 1 자료(고성룡 등, 2020 재인용)를 사용하여 프로그램의 사용 과정을 설명하고, 분석의 결과를 SNUDM의 결과(고성룡 등, 2020)와 비교해보고자 한다. 이 자료는 16명을 대상으로 어휘 판단 과제를 실시하고 반응시간을 측정된 것이며, 과제에는 고빈도(high frequency), 저빈도(low frequency), 극저빈도(very low frequency) 등의 3가지 빈도 조건과 유사 단어(pseudowords) 조건이 사용되었다. 참여자는 빈도 조건에서 각각 250 개의 단어를 판단하였고, 유사 단어는 750 개를 판단하였다. SNUDM은 CPU를 이용하여 무작위걷기 방법으로 확산과정을 근사한다. 그래픽처리장치를 이용하여 무작위걷기 방법으로 확산과정을 근사하는 것이 기존의 결과를 반복할 수 있는지 확인하고, CPU를 사용하는 것에 비해 더 빠르게 결과를 산출할 수 있는지 확인해보도록 한다. 또한 가상의 자료를 생성하여 모형이 매개변수를 복구할 수 있는지 확인하려 한다.

자료 파일

SNUDM-G의 분석에는 반응시간의 분위수 자료가 사용되며, 입력 파일은 'dataez.txt'라는 이름으로 저장되어야 한다. 입력 파일의 예는 그림 1과 같다. 반응시간 분위수 자료가 각 열에 참여자별로 반응(경계)마다 정리되어 있다. 자료의 행은 각각 참여자 번호, 조건, 반응('예': 1, '아니오': 0), 0.1 분위수, 0.3 분위수, 0.5 분위수, 0.7 분위수, 0.9 분위수, 반응수, 반응시간의 평균, 반응시간의 분산, 반응비율을 나타낸다. 제시된 자료는 참여자 1명의 4개의 조건에서의 '예' 반응과 '아니오' 반응에 해당하는 반응시간 분위수임을 확인할 수 있다. 각 조건은 유사 단어 조건, 고빈도 조건, 저빈도 조건, 극저빈도 조건을 나타낸다. 그림 1에서 첫 번째 조건인 유사 단어 조건의 경우 다른 조건과 달리 0('아니오') 반응의 자료가 먼저 나열되어 있다. SNUDM-G에서는 앞서 기술한대로 EZ-diffusion을 써서 표집률과 시작위치 그리고 비결정시간의 초깃값을 결정하는데, 이때 사용되는 자료가 조건의 첫 번째 행 즉 정반응의 반응시간 평균과 분산, 반응비율이다. 따라서 유사 단어 조건에서는 정반응인 0을

```

1 Pseudowords 0 0.492 0.556 0.613 0.691 0.884 696 0.661 0.003 0.928
1 Pseudowords 1 0.467 0.548 0.629 0.747 1.008 54 0.698 0.009 0.072
1 Highfrequency 1 0.453 0.501 0.542 0.591 0.710 243 0.571 0.001 0.972
1 Highfrequency 0 0.0 0.0 0.0 0.0 0.0 5 0.571 0.001 0.028
1 Lowfrequency 1 0.487 0.547 0.597 0.670 0.841 228 0.642 0.001 0.910
1 Lowfrequency 0 0.480 0.545 0.627 0.723 0.952 22 0.682 0.01 0.09
1 Verylowfrequency 1 0.497 0.565 0.632 0.717 0.912 201 0.683 0.002 0.804
1 Verylowfrequency 0 0.476 0.553 0.624 0.732 0.984 49 0.686 0.009 0.196
    
```

(그림 1) Ratcliff 등 (2004) 실험 1 자료(고성룡 등, 2020 재인용)

먼저 써준다.

실행

반응시간 분위수 자료 파일 ‘dataez.txt’를 준비하고 프로그램을 실행하면 먼저 참여자 수와 조건의 수를 묻는 질문이 나온다. Ratcliff 등 (2004)의 실험 1 자료를 사용할 경우에는 각각 1과 4를 입력한다. 다음으로 조건 간 어떤 매개변수를 비교할지 선택한다. 자극의 질이나 과제 난이도는 표집률과 관련이 있으므로 표집률만 비교하고 다른 매개변수는 조건마다 차이가 없다고 입력한다.

결과 파일

매개변수의 추정이 완료되면 ‘best.txt’와 ‘flog.txt’의 2가지 결과 파일이 생성된다. ‘best.txt’에는 참여자별 최적의 매개변수 추정치와 네이만 카이제곱 값이 저장되며, ‘flog.txt’에는 실행 시 입력한 선택 사항과 최적화 과정에서 찾은 매개변수 추정치 그리고 네이만 카이제곱 값 등이 저장된다. Ratcliff 등(2004)의 실험 1 자료를 SNUDM-G로 분석한 결과는 표 1에 정리되어 있다. v_1 , v_2 , v_3 , v_4 는 각각 유사 단어 조건, 고빈도 조건, 저빈도 조건, 극저빈도 조건에서의 표집률을 의미한다. SNUDM-G의 매개변수 추정 결과와 SNUDM의 결과를 비교하면, 경계 a 와 유사 단어 조건의 표집률 v_1 를 제외한 모든 매개변수에서 SNUDM-G의 추정치의 절댓값이 SNUDM의 추정치의 절댓값에 비해 컸다. SNUDM의 결과에 비해 SNUDM-G의 결과에서 a 와 z 사이의 거리는 줄고 z 와 0사이의 거리는 증가하였기 때문에 v_1 은 줄어든 반면 v_2 와 v_3 , v_4 는 증가하였다고 볼 수 있다. 조건 간 표집률의 절댓값 순서는 SNUDM과 SNUDM-G가 동일했다. 단어의 빈도가 낮아질수록 표집률의 크기가 줄어들었으며, 유사 단어 조건의 표집률 절댓값은 저빈도 조건의 표집률 절댓값보다는 작고 극저빈도 조건의 표집률 절댓값보다는 크게 추정되었다. 4 조건에 대한 참여자 1 명의 매개변수를 추정하는 데에 걸린 시간은 SNUDM이 708.22초, SNUDM-G가 5.64초로 GPU를 이용하였을 때의 계산 속도가 CPU를 이용하였을 때보다 빠르다고 할 수 있다.

〈표 1〉 Ratcliff 등 (2004) 실험 1의 매개변수 추정치

	a	z	s_z	T_{cr}	s_t	v_1	v_2	v_3	v_4	η
SNUDM-G	0.1307	0.0759	0.1237	0.5009	0.1584	-0.3154	0.7396	0.3830	0.2930	0.1566
SNUDM	0.1426	0.0743	0.0731	0.4216	0.0891	-0.3252	0.4518	0.3270	0.2728	0.1174
Ratcliff	0.110	0.056	0.004	0.435	0.159	-0.213	0.396	0.216	0.128	0.070

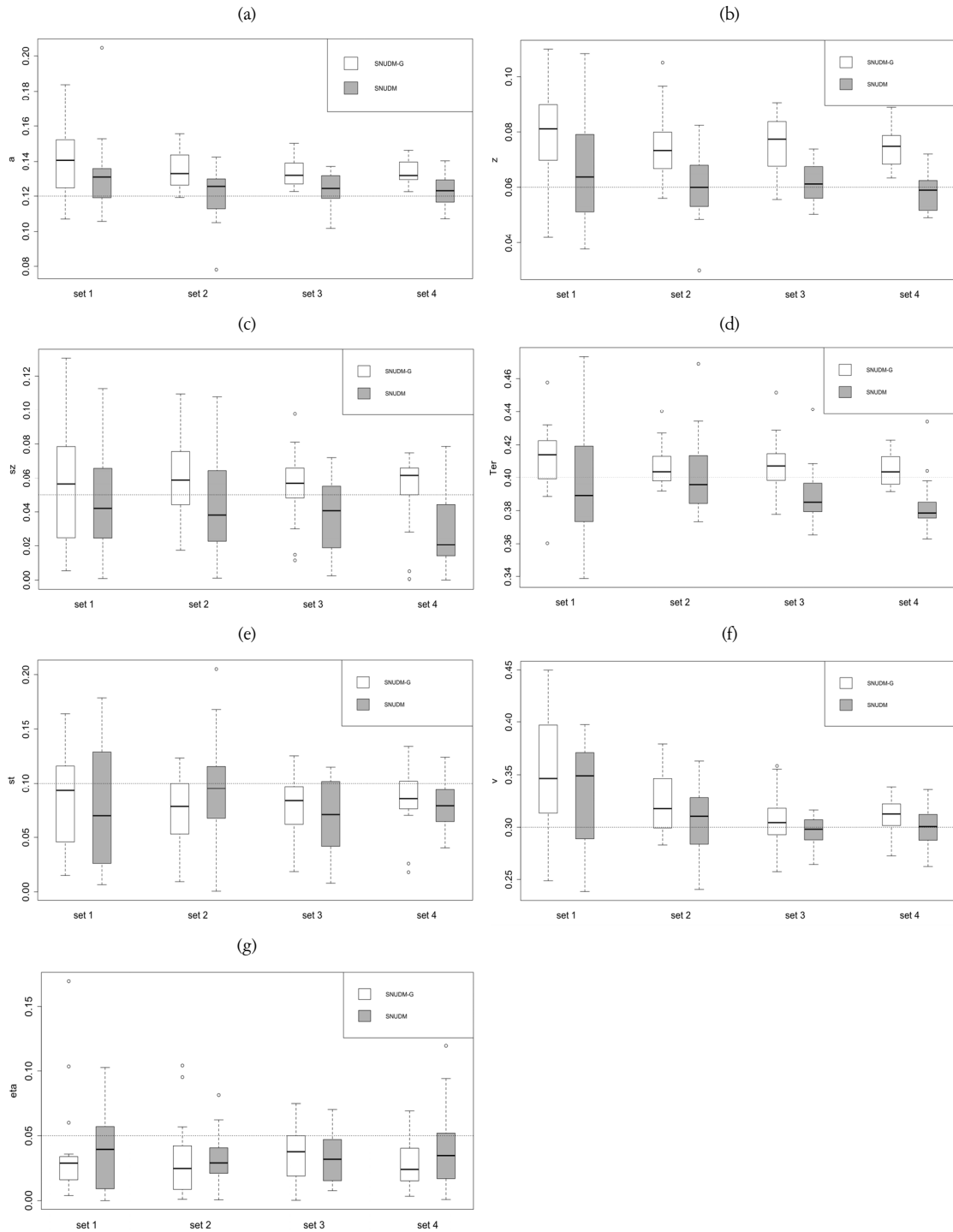
매개변수 복구

SNUDM-G가 매개변수를 복구할 수 있는지 확인하기 위해 특정 매개변수로 생성된 가상의 자료를 사용하여 SNUDM-G로 매개변수 추정을 하였다. 자료는 고성룡 등 (2020)에서 사용한 것과 동일하다. 자료의 생성에 사용된 매개변수의 값은 각각 경계 a 가 0.12, 시작위치 z 가 0.06, 표집률 v 가 0.3, 비결정시간 T_{er} 이 0.4였으며, 시행 간 변산도는 s_z 가 0.05, s_t 가 0.1 그리고 η 가 0.05였다. 자료는 20 명의 참여자가 각각 100 시행, 500 시행, 1000 시행, 10000 시행을 수행했다고 가정하여 생성되었다. 4 세트의 가상 자료에 대한 매개변수 추정 결과는 아래 표 2와 같다. 경계 a 와 시작위치 z , 비결정시간 T_{er} 그리고 표집률 v 의 평균은 모두 원래의 값보다 크게 추정되었으며, 시행 간 변산도를 나타내는 매개변수 중 s_z 의 평균도 마찬가지로 원래의 값보다 크게 추정되었다. 반면 s_t 와 η 의 평균은 원래의 값보다 작게 추정되었다.

〈표 2〉 가상 자료의 SNUDM-G 매개변수 추정치 평균과 표준편차

	a	z	s_z	T_{er}	s_t	v	η
set 1	0.1414 (0.0207)	0.0794 (0.0154)	0.0573 (0.0403)	0.4112 (0.0198)	0.0807 (0.0411)	0.3503 (0.0576)	0.0356 (0.0384)
set 2	0.1347 (0.0106)	0.0751 (0.0126)	0.0591 (0.0225)	0.4073 (0.0131)	0.0764 (0.032)	0.3252 (0.0306)	0.0303 (0.0289)
set 3	0.1328 (0.0079)	0.0753 (0.0104)	0.0551 (0.0202)	0.4074 (0.0152)	0.0779 (0.0315)	0.307 (0.0239)	0.0352 (0.0213)
set 4	0.1338 (0.0067)	0.0745 (0.0073)	0.0545 (0.0205)	0.4052 (0.01)	0.0855 (0.0269)	0.3119 (0.0171)	0.0281 (0.0177)

고성룡 등 (2020)에서 보고한 SNUDM의 가상 자료에 대한 매개변수 추정 결과와 SNUDM-G의 추정 결과를 비교하면 그림 2와 같다. 먼저 시행수가 증가할수록 두 모형의 추정 결과는 원래의 매개변수의 값에 다가가는 것으로 보인다. 다만 SNUDM-G의 결과에서 경계 a 와 시작위치 z , 비결정시간 T_{er} 그리고 표집률 v 가 모두 SNUDM의 결과보다 더 크게 추정되는 경향을 보였다. 경계 a 의 평균은 시행 수가 500, 1000, 10000 일 때 두 결과의 차이가 유의했다($t = 3.4733, p < .05$; $t = 3.2109, p < .05$; $t = 4.0194, p < .001$). 시작위치 z 의 평균은 4가지 자료에서 모두 유의한 차이가 있었고($t = 2.4578, p < .05$; $t = 4.02, p < .001$; $t = 4.6606, p < .001$; $t = 7.2018, p < .001$), 비결정시간 T_{er} 의 평균은 시



(그림 2) 가상 자료의 SNUDM-G와 SNUDM 매개변수 추정치

행 수가 1000 그리고 10000인 경우에 유의한 차이가 있었다($t = 3.5178, p < .01; t = 5.4885, p < .001$). 표집률 ν 의 평균은 모든 자료에서 유의한 차이가 발견되지 않았다. 시행 간 변산도 매개변수에서는 s_z 가 SNUDM-G의 결과에서 SNUDM의 결과보다 더 크게 추정된 경향을 확인할 수 있었다. 시행의 수가 1000 그리고 10000일 때 두 추정 결과의 평균 차이가 유의했다($t = 2.7189, p < .01; t = 4.0551, p < .001$).

결 론

본 연구에서는 인지 과제의 결과를 분석할 수 있는 확산모형 분석 도구 SNUDM-G를 제안하였다. 확산모형은 반응시간과 반응률을 모두 설명할 수 있고, 인지적 요인과 대응하는 매개변수를 가지고 있다는 장점에도 불구하고 수학적, 계산적 어려움으로 인해 사용에 제한이 있었다. 이 글에서는 특히 확산모형의 계산적 부담을 줄이기 위한 방법으로 그래픽처리장치를 이용할 것을 제안했다. SNUDM-G에서는 확산과정을 무작위걷기 방법으로 근사하는데, 이 과정에서 다량의 시뮬레이션 자료를 생성해야하는 부담을 줄이기 위해 2만 개의 무작위걷기 과정을 순차적으로 처리하지 않고 그래픽처리장치의 병렬처리 방식을 사용하였다. 같은 자료를 사용하여 SNUDM-G를 무작위걷기 시뮬레이션에 CPU를 사용하여 계산하도록 한 SNUDM과 비교해본 결과 몇몇의 매개변수에서 SNUDM-G의 추정 결과가 SNUDM의 추정 결과보다 다소 높게 나타났다. Ratcliff (2004)의 실험 1 자료를 사용하여 확산모형의 매개변수를 추정한 결과에서는 경계 a 와 유사 단어 조건의 표집률 ν_1 을 제외한 모든 매개변수에서 SNUDM-G의 추정 결과의 절댓값이 SNUDM의 추정 결과의 절댓값보다 더 높았다. 매개변수 복구에서도 표집률의 변산도 η 와 비결정시간의 변산도 s_t 를 제외한 모든 매개변수의 SNUDM-G 추정값이 SNUDM의 추정값보다 더 높은 경향을 보였다. 특히 경계 a 와 시작위치 z , 비결정시간 T_{er} 그리고 시작위치의 변산도 s_z 의 평균에서 SNUDM-G와 SNUDM의 유의한 차이를 발견할 수 있었다. 이 중 비결정시간 T_{er} 의 평균과 시작위치의 변산도 s_z 의 평균 차이는 SNUDM의 추정 결과가 자료 생성에 사용된 값보다 작은 것에 기인한 것으로 보인다. 따라서 이 2가지 매개변수의 경우, 유의미한 평균의 차이가 SNUDM-G의 결과가 SNUDM의 결과보다 정확도가 낮음을 의미한다고 보기 어렵다. 한편 경계 a 와 시작위치 z 는 SNUDM-G의 추정 결과와 SNUDM의 추정 결과가 모두 원래의 매개변수보다 크거나 유사하였다. 따라서 이 2가지 매개변수에서는 SNUDM-G의 추정 결과가 SNUDM의 추정 결과보다 정확도가 낮아 보인다. 이러한 결과는 그래픽처리장치를 사용하기 위한 CUDA에서의 실수 처리 방식의 정밀도나 난수 생성 방식 그리고 그래픽처리장치에서의 계

산 결과를 CPU와 동기화하는 부분 등의 한계에서 비롯되는 것이라 추측할 수 있다. 이러한 점을 해결하는 것이 SNUDM-G의 정확도를 향상시킬 수 있는 방법이 될 것이다. 또한 많은 메모리가 필요한 경우 그래픽처리장치의 아키텍처로 인해 스레드(thread) 당 사용할 수 있는 GPU RAM의 제약이 문제가 될 수 있다(Kuroiwa & Fukunaga, 2018). 이 글의 모형에서는 그래픽처리장치를 효율적으로 사용하기 위해 CUDA의 스트림(stream)을 사용하였으나, 이 방법 이외에 더 근본적인 처리 방법을 찾는 것이 과제라 할 수 있다. 그럼에도 불구하고 단어 조건 간의 SNUDM-G의 표집률 추정 결과는 SNUDM의 표집률 추정 결과와 동일한 조건 간의 차이를 보여주었다. 또한 SNUDM을 사용한 경우 1명의 자료를 처리하는 데 10분 이상이 소요되는 반면 SNUDM-G를 사용한 경우에는 6초 이내가 소요되었다. 즉 계산 속도 측면에서 SNUDM-G가 SNUDM보다 향상되었다는 것을 확인할 수 있었다. 특히 다수의 심리학 연구에서는 참여자의 수가 20명 혹은 그 이상인데, 20명의 자료를 SNUDM을 사용하여 처리하면 매우 단순하게 계산하였을 때 약 4시간 정도가 걸릴 것으로 예상할 수 있다. 반면 SNUDM-G를 사용하면 약 2분 정도가 걸릴 것이다. 그래픽처리장치를 사용한 방식이 확산모형으로 자료를 분석할 때 처리 효율을 극적으로 향상시켜 줄 것으로 보인다.

이 연구에서 제안한 분석 도구는 Ratcliff (1978)의 확산모형을 분석할 수 있는 프로그램으로 그래픽처리장치를 활용하여 계산 속도를 높인 것이다. Verdonck 등 (2016)은 그래픽처리장치를 이용한 확산모형의 속도 개선이 Ratcliff가 제안한 기본적인 확산모형뿐만 아니라 3개 이상의 선택지가 경쟁을 하는 LCA(leaky competing accumulator) 모형(Usher & McClelland, 2001) 등에도 적용 가능함을 보였다. 또한 Voss 등 (2019)은 확산과정의 노이즈가 정규분포 이외의 분포를 따르는 경우를 가정하여 레비 비행 모형을 제안하였으며, 이 모형이 특히 시간이 짧은 오반응을 분석할 때 유용할 수 있음을 주장하였다. 이후의 연구에서는 그래픽처리장치를 이용하여 위와 같은 확산모형의 다양한 변형을 다룰 수 있는 확산모형 분석도구를 고안해볼 수 있을 것이다.

따라서 SNUDM-G는 그래픽처리장치를 이용하여 확산모형 분석을 시도할 때 CUDA에서의 실수 처리나 난수 생성 방식과 동기화 방식 등에 대한 개선이 필요함을 보여줌과 동시에 CPU를 이용하여 확산모형 분석을 하는 것보다 빠른 계산 속도로 조건 간 매개변수의 차이를 확인할 수 있게 해준다. 이는 다양한 인지 과제에 대해 보다 효율적인 확산모형의 분석을 가능하게 할 것이다. 또한 양자택일 과제를 다루는 확산모형뿐만 아니라 다양한 확산모형에 그래픽처리장치를 사용하여 효율적인 분석할 수 있는 가능성을 제시한다는 점에서 의의가 있다.

참고문헌

- 고성룡, 주혜리, 이다정. (2020). 확산모형 분석도구: SNUDM. *인지과학*, 31(1), 1-23.
- Arnold, N. R., Broder, A., & Bayen, U. J. (2015). Empirical validation of the diffusion model for recognition memory and a comparison of parameter-estimation methods. *Psychological Research*, 79(5), 882-898.
- Brown, S., Ratcliff, R., & Smith, P. L. (2006). Evaluating methods for approximating stochastic differential equations. *Journal of Mathematical Psychology*, 50, 401-410.
- Feller, W. (1968). *An introduction to probability theory and its applications*. New York: Wiley.
- Kuroiwa, R., & Fukunaga, A. (2018). Batch random walk for GPU-based classical planning. In *Twenty-Eighth International Conference on Automated Planning and Scheduling*.
- Lerche, V., & Voss, A. (2017). Experimental validation of the diffusion model based on a slow response time paradigm. *Psychological Research*, 83(6), 1194-1209.
- Luersen, M. A., Le Riche, R., & Guyon, F. (2004). A constrained, globalized, and bounded Nelder-Mead method for engineering optimization. *Structural and Multidisciplinary Optimization* 27(1), 43-54.
- Nelder, J. A., & Mead, R. (1965). A simplex method for function minimization. *Computer Journal*, 7, 308-313.
- Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, 85, 59-108.
- Ratcliff, R. (1981). A theory of order relations in perceptual matching. *Psychological Review*, 88, 552-572.
- Ratcliff, R. (2014). Measuring psychometric functions with the diffusion model. *Journal of Experimental Psychology: Human Perception and Performance*, 40, 870-888.
- Ratcliff, R., Gomez, P., & McKoon, G. (2004). A diffusion model account of the lexical decision task. *Psychological Review*, 111(1), 159-182.
- Ratcliff, R., & McKoon, G. (2008). The diffusion decision model: Theory and data for two-choice decision tasks. *Neural Computation*, 20, 873-922.
- Ratcliff, R., & Rounder, J. N. (1998). Modeling response times for two-choice decisions. *Psychological Science*, 9, 347-356.
- Ratcliff, R., Smith, P. L., Brown, S. D., & McKoon, G. (2016). Diffusion decision model: Current issues and history. *Trends in Cognitive Sciences*, 20, 260-281.
- Ratcliff, R., Thapar, A., & McKoon, G. (2001). The effects of aging on reaction time in a signal detection task. *Psychology and Aging*, 16, 323-341.

- Ratcliff, R., Thapar, A., & McKoon, G. (2010). Individual differences, aging, and IQ in two-choice tasks. *Cognitive Psychology*, 60, 127-157.
- Ratcliff, R., & Tuerlinckx, F. (2002). Estimating parameters of the diffusion model: Approaches to dealing with contaminant reaction times and parameter variability. *Psychonomic Bulletin & Review*, 9, 438-481.
- Ratcliff, R., Vand Zandt, T., & McKoon, G. (1999). Connectionist and diffusion models of reaction time. *Psychological Review*, 106, 261-300.
- Starns, J. J., & Ratcliff, R. (2010). The effects of aging on the speed-accuracy compromise: Boundary optimality in the diffusion model. *Psychology and Aging*, 25, 377-390.
- Starns, J. J., & Ratcliff, R. (2012). Age-related differences in diffusion model boundary optimality with both trial-limited and time-limited tasks. *Psychonomic Bulletin & Review*, 19, 139-145.
- Usher, M., & McClelland, J. L. (2001). The time course of perceptual choice: The leaky, competing accumulator model. *Psychological Review*, 108(3), 550.
- Verdonck S., Meers, K., & Tuerlinckx, F. (2016). Efficient simulation of diffusion-based choice RT models on CPU and GPU. *Behavior Research Methods*, 48(1), 13-27.
- Voss, A., Lerche, V., Mertens, U., & Voss, J. (2019). Sequential sampling models with variable boundaries and non-normal noise: A comparison of six models. *Psychonomic Bulletin & Review*, 26(3), 813-832.
- Voss, A., Rothermund, K., & Voss, J. (2004). Interpreting the parameters of the diffusion model: An empirical validation. *Memory & Cognition*, 32, 1206-1220.
- Voss, A., & Voss, J. (2007). Fast-Dm: A free program for efficient diffusion model analysis. *Behavior Research Methods*, 39(4), 767-775.
- Voss, A., & Voss, J. (2008). A fast numerical algorithm for the estimation of diffusion model parameters. *Journal of Mathematical Psychology*, 52(1), 1-9.
- Wagenmakers, E. J., Van Der Maas, H. L., & Grasman, R. P. (2007). An EZ-diffusion model for response time and accuracy. *Psychonomic Bulletin & Review*, 14(1), 3-22.

1차 원고 접수: 2022. 01. 10

1차 심사 완료: 2022. 08. 29

2차 원고 접수: 2022. 09. 08

2차 심사 완료: 2022. 09. 13

최종 게재 확정: 2022. 09. 16

(Abstract)

Analysis tool for the diffusion model using GPU: SNUDM-G

Dajung Lee¹⁾ Hyosun Lee²⁾ Sungryong Koh¹⁾²⁾

¹⁾Department of Psychology, Seoul National University

²⁾Cognitive Program, Seoul National University

In this paper, we introduce the SNUDM-G, a diffusion model analysis tool with improved computational speed. Although the diffusion model has been applied to explain various cognitive tasks, its use was limited due to computational difficulties. In particular, SNUDM(Koh et al., 2020), one of the diffusion model analysis tools, has a disadvantage in terms of processing speed because it sequentially generates 20,000 data when approximating the diffusion process. To overcome this limitation, we propose to use graphic processing units(GPU) in the process of approximating the diffusion process with a random walk process. Since 20,000 data can be generated in parallel using the graphic processing units, the estimation speed can be increased compared to generating data through sequential processing. As a result of analyzing the data of Experiment 1 by Ratcliff et al. (2004) and recovering the parameters with SNUDM-G using GPU and SNUDM using CPU, SNUDM-G estimated slightly higher values for certain parameters than SNUDM. However, in term of computational speed, SNUDM-G estimated the parameters much faster than SNUDM. This result shows that a more efficient diffusion model analysis for various cognitive tasks is possible using this tool and further suggests that the processing speed of various cognitive models can be improved by using graphic processing units in the future.

Key words : diffusion model, GPU, random walk