

## 빈발도와 가중치를 적용한 엣지 디바이스의 최적 이동패턴 추출

이연식<sup>1\*</sup> · 장민석<sup>1</sup>

### Extraction of Optimal Moving Patterns of Edge Devices Using Frequencies and Weights

YonSik Lee<sup>1\*</sup> · MinSeok Jang<sup>1</sup>

<sup>1\*</sup>Professor, School of Computer Info. & Comm., Kunsan National University, Kunsan, 54150 Korea

#### 요약

최근 클라우드 환경의 서비스 지연문제 해결을 위하여 응용서비스의 사용자 근접성 확보 및 계산 오프로딩을 위한 FEC (Fog/Edge Computing) 패러다임에 대한 연구가 활발하다. 엣지 디바이스 (이동객체)들의 동적 위치변화 패턴 예측방법은 FEC 환경에서 컴퓨팅 리소스의 효율적 분배 및 배치를 위하여 중요한 역할을 한다. 본 논문은 엣지 디바이스들의 이동 빈발패턴에 대한 지도의 임계값을 적용하여 선택된 경로들을 대상으로 임의의 가중치 (거리, 시간, 혼잡도)를 추가적으로 적용한 최적 이동패턴 추출방법을 제안한다. 실험을 통하여 제안 알고리즘은 빈발도만을 적용한 OPE\_freq [8] 알고리즘과 A\* 및 Dijkstra 알고리즘 등과 비교한 결과, 수행시간과 노드 접근횟수를 감소시키고 보다 정확한 경로를 추출함을 알 수 있다.

#### ABSTRACT

In the cloud computing environment, there has been a lot of research into the Fog/Edge Computing (FEC) paradigm for securing user proximity of application services and computation offloading to alleviate service delay difficulties. The method of predicting dynamic location change patterns of edge devices (moving objects) requesting application services is critical in this FEC environment for efficient computing resource distribution and deployment. This paper proposes an optimal moving pattern extraction algorithm in which variable weights (distance, time, congestion) are applied to selected paths in addition to a support factor threshold for frequency patterns (moving objects) of edge devices. The proposed algorithm is compared to the OPE\_freq [8] algorithm, which just applies frequency, as well as the A\* and Dijkstra algorithms, and it can be shown that the execution time and number of nodes accessed are reduced, and a more accurate path is extracted through experiments.

**키워드** : 포그/엣지 컴퓨팅, 엣지 디바이스, 최적 이동패턴, 시공간 가중치, 자원 배치

**Keywords** : Fog/Edge Computing, Edge Device, Optimal Moving Pattern, Spatio-temporal Weights, Resource Placement

Received 20 March 2022, Revised 30 March 2022, Accepted 14 April 2022

\* Corresponding Author YonSik Lee (E-mail: yslee@kunsan.ac.kr, Tel:+82-63-469-4553)

Professor, School of Computer Info. & Comm., Kunsan National University, Kunsan, 54150 Korea

Open Access <http://doi.org/10.6109/jkiice.2022.26.5.786>

print ISSN: 2234-4772 online ISSN: 2288-4165

© This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.  
Copyright © The Korea Institute of Information and Communication Engineering.

## I. 서론 및 제안

최근 사용자 중심의 차세대 컴퓨팅 서비스를 위한 FEC 패러다임에 대한 연구가 활발하다 [1, 2]. FEC 환경에서 시공간 속성이 지속적으로 변화하는 엷지 디바이스들의 응용 서비스 처리를 위하여, 디바이스들의 위치 변화 패턴 예측을 통한 컴퓨팅 리소스의 효율적 배치방법이 요구된다 [2-4]. 이를 위하여 선행연구로 임의의 시공간 환경에서 응용 서비스를 요구하는 이동객체들의 이동패턴을 빈발도 기반으로 분석하여 최적 이동패턴 추출 알고리즘을 제시하였으나 [5,8], 빈발도만을 적용한 이동패턴은 최적일 가능성은 높지만 최소비용의 경로임은 보장할 수 없다 [6]. 따라서 본 논문에서는 빈발 패턴 지도도의 임계값을 적용한 산출 경로들을 대상으로 임의의 가중치(거리, 시간, 혼잡도)를 추가적으로 적용하여 최소비용을 소요하는 최적경로 추출방법을 제안한다. 이는 경험적 가중치를 적용하는 A\*알고리즘의 경로추출 성능의 비교분석 결과를 참고하여 [7,8], 다음과 같은 가정을 기반으로 한다.

- [가정 1] 이동객체의 빈발 지도도가 낮은 이동패턴은 최적일 확률이 낮으며, 최대 빈발도 만을 적용한 이동패턴은 시공간 환경변수나 사용자 성향으로 인하여 최소비용을 소요하는 최적 이동패턴이 될 가능성이 낮다.
- [가정 2] 빈발 지도도 임계값 이상의 이동패턴들에 가중치(거리, 시간, 혼잡도)를 적용한 비용이 최소인 패턴이 최적 이동패턴이 될 가능성이 높다.

최적 이동패턴은 시공간 속성으로 일반화된 이동시퀀스 집합으로부터 주어진 최소지도도 이상이면서 적용 가중치가 최소인 이동시퀀스를 이용하여 추출한다. 기존 Apriori 계열 알고리즘들의 빈발 이동패턴 추출 과정은 k-시퀀스 집합에서 (k+1)-시퀀스 집합을 찾을 때 패스 단위의 반복 접근법을 사용한다 [7]. 선행연구[8]에서의 제안 방법은 패스 단위의 반복 접근법을 사용하지만, 후보 시퀀스 집합에서 빈발 2-node 시퀀스를 찾을 때 후보 시퀀스들의 차수가 단계적으로 줄어드는 특징이 있다. 본 논문은 이러한 후보 시퀀스 집합으로부터 빈발 2-node 시퀀스들의 순차적 조합을 통하여 빈발 이동패턴을 추출하고, 가중치를 추가 적용하여 보다 정확한 최적 이동패턴을 추출하는 방법을 제안한다.

본 논문의 구성은 II장에서 적용 데이터 및 가중치를 산출하고, III장에서 최적 이동패턴 추출 알고리즘과 추출과정을 제시한다. IV장에서 실험을 통하여 성능을 평가하고, 이를 기반으로 V장에서 결론을 제시한다.

## II. 적용 데이터 및 가중치 산출

### 2.1. 최적 이동패턴 추출을 위한 데이터

AoE (App. on Edge/Mobile Device)의 이동패턴 추출은 이동경로 탐사를 기반으로 수행하므로, AoE의 이동이력데이터는 최하위 수준의 시공간 속성으로 일반화된 시퀀스 데이터로 표현한다. 따라서 AoE의 이동패턴 추출을 위해서는 AoE의 이동이력데이터에 시간간격 제약조건을 적용하여 이동시퀀스로 생성하고, 데이터 일반화를 통해 이들을 일반화된 시퀀스로 변환하여 사용한다. 다음 그림 1은 AoE가 이동 가능한 공간속성으로 표현된 도로 네트워크를 모델링한 것으로, 공간의 최상위 레벨은  $R = \{R1, R2, R3\}$ 이며,  $R1 = \{A1, A2, \dots, A1\}$ ,  $R2 = \{B1, B2, \dots, Bm\}$ ,  $R3 = \{C1, C2, \dots, Cn\}$  이고, 각 A, B, C는 최하위 레벨인 각 링크를 연결하는 노드들의 집합으로 AoE의 위치 샘플링에 사용한다.

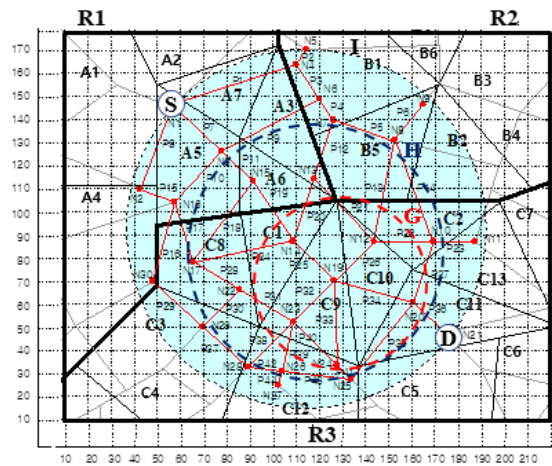


Fig. 1 Road network expressed by spatial attributes

표 1은 그림 1의 도로 네트워크상에서 AoE의 이동이력데이터로부터 일반화 및 요약을 통해 생성한 이동시퀀스 데이터 집합이다.

**Table. 1** Example of moving sequence data set

OID	Sequence
MO1	<N1 N3 N16 N17 N22 N23 N24 N25 N20 N21>
MO2	<N1 N3 N15 N14 N23> <N23 N24 N25 N20 N21>
MO3	<N1 N2 N16 N17 N22 N23 N24 N25 N20 N21>
MO4	<N1 N4 N6 N7 N8 N10 N20 N21>
MO5	<N1 N4 N6 N7 N13 N12 N19 N20 N21>
MO6	<N1 N3 N16 N30 N29> <N29 N22 N23 N19 N20 N21>
MO7	<N1 N4 N6 N7 N8 N12 N10 N20 N21>
MO8	<N1 N2 N16 N30 N29 N28> <N28 N23 N19 N20 N21>

**2.2. 최적 이동패턴 추출을 위한 가중치**

[가정 1]과 [가정 2]를 근거로, 빈발도와 가중치를 동시에 적용한 최적 이동패턴 추출은 AoE의 이동이력데이터베이스 (AoE\_DB) 중 지정된 공간제약 (출발점과 도착점)을 포함하는 최소 원형범위에 포함되는 이동패턴들을 탐색대상으로 설정한다. 시간제약과 최소지지도 (지지도 임계값)가 주어질 때, 이들을 만족하는 이동시퀀스 집합으로부터 빈발 부분시퀀스를 추출하고 그들 중 적용 가중치가 최소인 시퀀스들의 연결리스트를 최적 이동패턴으로 결정한다.

각 경로에 적용하는 가중치는 보정시간 ( $c_{vt}$ )과 보정거리 ( $c_d$ )를 적용하며, 보정시간에는 혼잡도 (CR)를 부가적으로 적용한다. 보정거리는 주어진 공간범위 내에서 가장 긴 경로 (4.5km)를 1로 설정하여 정규화한 비교값을 사용하고, 보정시간은 보정거리에 혼잡도를 곱한 후 가장 큰 값을 1로 설정하여 정규화한 값을 사용한다. 또한 실제 가중치를 적용할 경우, 상황에 따라 임의 설정이 가능한 각 가중치별 영향지수 (effect factor)인  $ef_{vt}$ 와  $ef_d$ 를 동시에 적용하며, 본 논문에서는 실험을 위하여  $ef_{vt} = 0.7$ ,  $ef_d = 0.3$ 을 적용한다. 이동 정보들의 가중치는 상대적 효율성을 백분율로 표시한다.

적용 가중치는 식 (1)과 같이 산출하며, 전체 경로들의 산출된 적용 가중치는 표 2와 같다.

$$\begin{aligned}
 \text{Applied Weight (AW)} &= a_{vt} + a_d & (1) \\
 a_{vt} &= c_{vt} * ef_{vt}, a_d = c_d * ef_d \\
 c_{vt} &= c_d * CR, c_d = d / 4.5(km)
 \end{aligned}$$

**Table. 2** Applied weights calculated by applying the effect factors

Path	weight			Path	weight		
	a_vt	a_d	AW		a_vt	a_d	AW
P1	0.31	0.30	0.61	P23	0.18	0.09	0.27
P2	0.05	0.05	0.10	P24	0.70	0.23	0.93
P3	0.11	0.11	0.22	P25	0.39	0.13	0.52
P4	0.07	0.07	0.14	P26	0.39	0.13	0.52
P5	0.15	0.15	0.30	P27	0.45	0.15	0.60
P6	0.11	0.11	0.22	P28	0.25	0.12	0.37
P7	0.17	0.17	0.34	P29	0.16	0.15	0.31
P8	0.21	0.20	0.41	P30	0.25	0.12	0.37
P9	0.25	0.25	0.50	P31	0.43	0.14	0.57
P10	0.16	0.15	0.31	P32	0.41	0.13	0.54
P11	0.20	0.10	0.30	P33	0.59	0.19	0.78
P12	0.29	0.14	0.43	P34	0.55	0.18	0.73
P13	0.65	0.21	0.86	P35	0.70	0.23	0.93
P14	0.48	0.23	0.71	P36	0.35	0.11	0.46
P15	0.09	0.09	0.18	P37	0.14	0.13	0.27
P16	0.18	0.18	0.36	P38	0.45	0.15	0.60
P17	0.29	0.14	0.43	P39	0.35	0.11	0.46
P18	0.47	0.23	0.70	P40	0.41	0.13	0.54
P19	0.47	0.15	0.62	P41	0.12	0.06	0.18
P20	0.45	0.15	0.60	P42	0.15	0.07	0.22
P21	0.59	0.19	0.78	P43	0.31	0.15	0.46
P22	0.41	0.13	0.54	P44	0.10	0.05	0.15

**III. 빈발도와 가중치를 적용한 최적 이동패턴 추출**

**3.1. 빈발도와 가중치를 적용한 최적 이동패턴 추출 알고리즘**

빈발도와 가중치를 적용한 최적 이동패턴 추출은 최소지지도 이상의 최대 빈발 2-node 시퀀스들 중 적용 가중치가 최소인 2-node 시퀀스를 추출하는 방법으로, 빈발도만을 적용한 기존연구 [8]와 특징적 차이가 있다.

제안방법은 첫째, AoE\_DB로부터 시공간 제약 조건을 만족하는 후보 시퀀스 집합을 추출하고, 그들로부터 2-node 시퀀스 집합을 추출한다. 추출된 2-node 시퀀스들에 대해 최소지지도와 시간, 거리 및 혼잡도에 대한

가중치를 적용하여 목적지 까지 구해진 2-node 시퀀스들의 선형리스트를 최적 이동패턴으로 결정한다. 최적 이동패턴을 추출하기 위하여, 먼저 limited Data Set 함수에서 시공간 한정자와 공간 제약 조건을 이용하여 AoE\_DB로부터 AoE의 이력데이터를 추출하고, 이들을 SEQ\_Extract 함수에서 시간간격 조건을 적용한 이동시퀀스들을 생성한다. 생성된 이동시퀀스들을 대상으로 시간 및 공간 속성을 일반화하고, 일반화된 이동시퀀스를 대상으로 최적 이동패턴을 추출한다.

다음 그림 2는 빈발도와 가중치를 선택적으로 적용할 수 있도록 구현한 OPE\_FW 알고리즘이다.

```

Input : AoE-DB, Cs (Spatial Constraint),
S (Start node), currN (Current node),
D (Destination node),
max_gap (Temporal Constraint),
min_sup (Threshold of Minimum Support factor),
G-Mseq (Generalized Moving Sequence),
ToW (Threshold of Weight), type (applied type)
Output : Optimal Moving Pattern (OMP_list)
Procedure OPE_FW (AoE-DB, Cs, S, D, max_gap, min_sup)
Begin
  PrevA = null;
  // Set of historical data satisfied Cs
  Drt = limited Data Set (AoE-DB, Cs, S, D);
  // Extraction of set of moving sequences from Drt
  Dseq = SEQ_Extract (Drt, max_gap);
  // Data Generalization
  // Extraction of optimal moving pattern
Begin
  OMP_list = null;
  nextN = S;
  while (nextN ≠ D) {
    add nextN to OMP_list;
    Begin
      subD = Sequence containing currN and D in
      G-Mseq;
      for each moving sequence S ∈ subD
        c = subset(currN, S);
        if c ⊆ Cs then add c to Cs;
        c.count++;
      End for
      If (type==1) {
        cmax_sup = 2-node sequence with maximum support
        factor from Cs;

```

```

      nextN = new node unconnected with currN in cmax_sup;
    }
    Else {
      Freqseq = 2-node sequence with support factor(≥
      minsup) from Cs;
      cmin_weight = 2-node sequence with threshold(≥ToW)
      from Freqseq;
      nextN = new node unconnected with currN in cmin_weight
    };
    Return nextN;
  }
  End
  add D to OMP_list;
End
Return OMP_list;
End

```

Fig. 2 OPE\_FW algorithm

### 3.2. 최적 이동패턴 추출을 위한 규칙 및 추출 과정

OPE\_FW 알고리즘을 이용한 최적 이동패턴 추출과정을 보이기 위하여 AoE\_DB로부터 시공간제약 조건을 만족하는 이동시퀀스 집합 (표 1)을 사용한다. 또한 빈발도만을 적용한 방법 [8]에 가중치를 추가적으로 적용하기 위하여 다음과 같은 규칙들을 정의한다.

- [규칙 1] 최소지지도(min<sub>sup</sub>) 이상의 경로들 중 적용가중치가 최소인 경로를 선택한다.
- [규칙 2] 비교 경로가 없을 경우에는 최소지지도와 관계없이 경로를 선택한다.
- [규칙 3] 경로들의 빈발 지지도가 주어진 최소지지도보다 작을 경우에는 가중치가 최소인 경로를 선택한다.
- [규칙 4] 경로들의 적용가중치가 동일할 경우에는 빈발지지도가 최대인 경로를 선택한다.

그림 3은 제안 OPE\_FW 알고리즘에 표 1의 이동시퀀스 집합, 표 2의 적용 가중치 및 [규칙 1]~[규칙 4]를 적용하여 최적 이동패턴 추출 과정의 예를 보인다.

추출된 최적 이동패턴은 다음과 같다.

<N1-N3-N16-N30-N29-N28-N23-N24-N25-N20-N21>

AoE의 위치기반 서비스를 위한 제안 방법의 시스템은 이동시퀀스 추출 모듈, 데이터 일반화 모듈 및 최적 이동패턴 추출 모듈로 구성한다. 구성 모듈들 간의 동작과정은 그림 4와 같다.

Start node	Path No.	to Node	Min_sup (≥ 0.2)	Applied Weight	Selected node
N1	P8	N2	0.25	0.41	N3
	P7	N3	0.375	0.34	
	P1	N4	0.375	0.61	
<b>Applied Rule 1</b>					
N3	P10	N16	0.25	0.31	N16
	P11	N15	0.125	0.30	
<b>Applied Min_sup of Frequency</b>					
N16	P17	N17	0.25	0.43	N30
	P16	N30	0.25	0.36	
<b>Applied weight</b>					
N30	P29	N29	0.25	0.31	N29
<b>Applied Rule 2</b>					
N29	P30	N22	0.125	0.37	N28
	P37	N28	0.125	0.27	
<b>Applied Rule 3</b>					
N28	P38	N23	0.125	0.6	N23
<b>Applied Rule 2</b>					
N23	P40	N24	0.375	0.54	N24
	P32	N29	0.25	0.54	
<b>Applied Rule 4</b>					
N24	P41	N25	0.375	0.18	N25
<b>Applied Rule 2</b>					
N25	P35	N20	0.375	0.93	N20
<b>Applied Rule 2</b>					
N20	P36	N21	1.0	0.46	N21
<b>Applied Rule 2</b>					

Fig. 3 Extraction of the optimal moving patterns based on frequency and weights

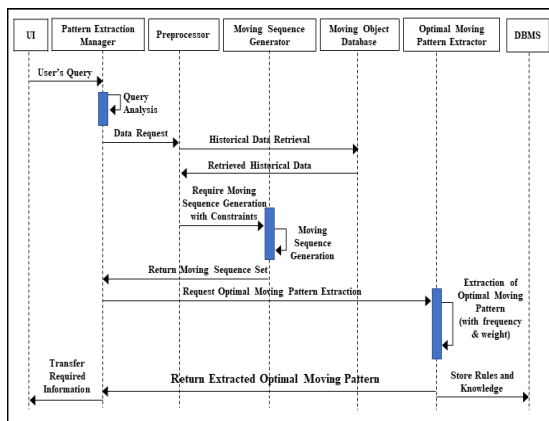


Fig. 4 Sequence diagram for optimal moving pattern extraction

## IV. 실험 및 성능분석

### 4.1. 실험 환경

본 장에서는 OPE\_FW 알고리즘의 최적 이동패턴 추출의 성능을 OPE\_freq, A\* 및 Dijkstra 알고리즘과 비교하여 평가한다. 실험은 알고리즘의 수행시간, 접근노드 수, 정확성을 기준으로 수행하였으며, A\* 알고리즘은 경험적 가중치에 따라 성능 차이가 크게 발생하므로 두 개의 구간범위 (A\*-a: 5~10Km, A\*-b: 15~20Km)로 구분하여 실험하였다. 실험은 Windows x64 OS에서 Eclipse IDE와 JDK 8, Oracle 10g 버전 데이터베이스 기반 환경에서 수행하였으며, AoE의 출발지와 목적지를 임의로 변경하여 각각 120, 240, 360, 480번의 경로탐색을 실행하여 평균값을 비교 평가하였다.

### 4.2. 실험 결과 평가

#### 4.2.1. 실행 시간 비교 결과

그림 5는 동일 시공간 범위에서 최적 이동패턴 추출을 위한 OPE\_FW 알고리즘의 수행시간을 OPE\_freq, A\* 및 Dijkstra 알고리즘들과 비교한 결과이다.

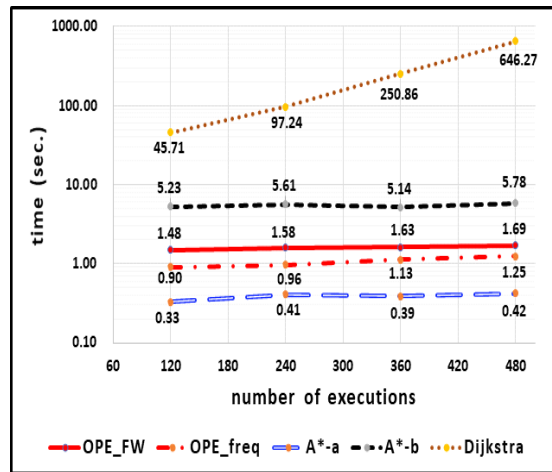


Fig. 5 Result of the execution time comparison

OPE\_FW 알고리즘은 OPE\_freq 알고리즘에 비하여 가중치에 대한 추가 연산시간이 더 소요되며, 경험적 가중치에 따른 직진성으로 인하여 구간범위가 클수록 수행시간은 길어지는 A\* 알고리즘에 비하여 구간범위의 크기에 따라 수행시간 성능에 차이가 있음을 보인다. 구간범위 내의 모든 노드를 탐색하는 Dijkstra 알고리즘과

는 매우 큰 차이가 있음을 알 수 있다. 수행횟수를 기준으로 한 비교에서 OPE\_FW 알고리즘은 Dijkstra 알고리즘을 제외한 다른 알고리즘들과 유사하게 연산시간 성능은 거의 비슷하였다.

4.2.2. 노드 접근횟수 비교 결과

최적경로 추출에서 노드 접근횟수는 경로 선택 여부를 결정하는 시간으로 인하여 알고리즘 성능에 밀접한 영향을 준다. OPE\_FW 알고리즘은 OPE\_freq 알고리즘과 마찬가지로 빈발도가 상대적으로 적은 노드들을 매 단계에서 제외(이력데이터로부터 빈발도가 높은 패턴들만 추출)하므로 다른 알고리즘에 비하여 상대적으로 유의미하게 적은 노드 접근횟수를 가진다. OPE\_FW 알고리즘은 최소 지도도를 적용한 빈발패턴을 대상으로 적용 가중치 산출을 위한 추가적인 노드 접근이 요구된다. A\* 알고리즘은 수행시간 평가와 마찬가지로 구간범위가 클수록 알고리즘의 직진성으로 인하여 접근 노드 수가 비효율적으로 증가함을 알 수 있다. Dijkstra 알고리즘은 도로 네트워크의 토폴로지에 따라 차이가 있지만 최적경로를 구성하는 각 노드로부터 접근 대상이 되는 최대 노드 수는 (n-1)개이므로 시간 복잡도가 O(n<sup>2</sup>)이 되어 상대적으로 비효율적임을 알 수 있다.

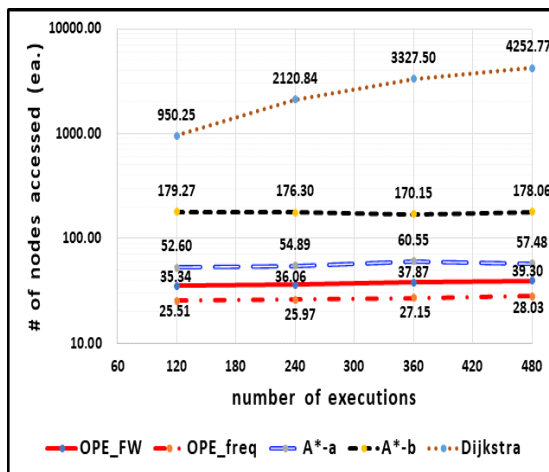


Fig. 6 Result of comparing the average number of access nodes

4.2.3. 정확도 비교 결과

다음 그림 7은 최적경로를 가장 정확하게 추출하는 Dijkstra 알고리즘과 정확도를 비교 평가한 결과이다.

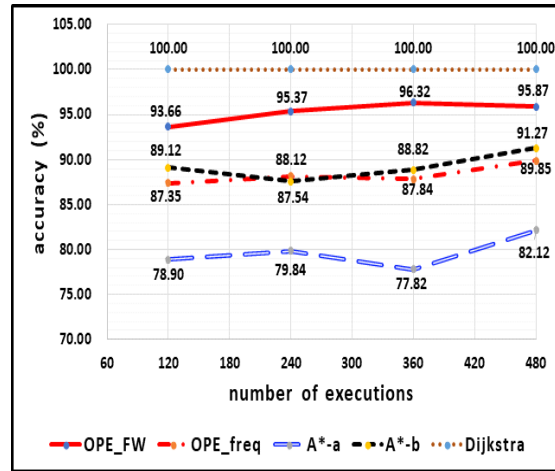


Fig. 7 Result of the accuracy comparison

경험적 가중치를 적용하는 A\* 알고리즘은 탐색 범위가 작을수록 정확성이 낮아지며, 경험적 가중치를 크게 하여 많은 노드들을 탐색하여 정확도를 높게 되면 수행시간이 크게 증가하는 문제가 있다. 따라서 경험적 가중치의 설정이 수행능력 평가에 주요 요인이 된다. OPE\_freq 알고리즘은 A\* 알고리즘들과 비슷한 정확도를 가지지만 상대적으로 빠른 수행시간으로 최적경로를 추출한다. OPE\_FW 알고리즘은 빈발도와 가중치를 동시에 적용함으로써, Dijkstra 알고리즘의 최적경로에 근접한 정확도를 보장하며 소요비용 측면에서도 최적인 경로를 추출할 수 있다.

V. 결론

본 논문은 FEC 환경에서 이동에이전트 기반의 계산 오프로딩의 효율성 제고를 위한 선행연구로서, 복잡한 시공간 제약을 적용한 AoE들의 최적 이동패턴을 추출하는 알고리즘을 제안하고, 그의 수행능력을 비교 평가하였다. 제안 알고리즘인 OPE\_FW 알고리즘은 이동객체의 시공간 속성 일반화와 이동패턴의 빈발도와 가중치 (거리, 선호도, 혼잡도)를 동시에 적용하여 데이터베이스 접근 횟수, 수행시간 및 소요 메모리를 감소시킬 뿐 아니라, 최적경로를 기존 알고리즘들에 비하여 정확하게 산출하는 특징을 가진다.

실험을 통하여 OPE\_FW 알고리즘을 OPE\_freq 알고

리즘 [8], A\* 및 Dijkstra 알고리즘과 수행시간, 접근노드 수 및 정확도를 기준으로 성능을 평가한 결과에서, OPE\_FW 알고리즘은 가중치를 산출 적용함으로써 OPE\_freq 알고리즘에 비하여 수행시간이 다소 증가할 뿐 다른 성능은 훨씬 우수함을 보였다. 구간범위가 커질수록 처리시간이 가파르게 증가하는 A\* 알고리즘과의 비교에서도 작은 구간범위에서의 수행시간은 다소 길지만 나머지 평가기준에 대해서는 우수함을 보였다. 특히 제안 알고리즘은 완벽한 정확도를 갖는 Dijkstra 알고리즘의 성능을 기준으로 OPE\_freq 알고리즘과 A\* 알고리즘들 보다 우수한 성능을 보였다.

본 논문에서 제안한 능동적인 최적 이동패턴 추출방법은 AoE의 시공간적 동적 특성이 요구되는 응용에 적용 가능하므로, 다양한 위치기반 서비스뿐만 아니라 FEC 환경에서 효율적 계산 오프로딩을 위한 컴퓨팅 리소스 배치 및 할당 시스템에 유용하게 적용이 가능하며, 부분 오프로딩 알고리즘 개발 및 이동객체의 기지국간 이동 시 발생하는 핸드오버 처리에 응용이 가능하다.

### ACKNOWLEDGEMENT

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (No. 2021R1F1A1047768) and a grant (22RITD-C161698-02) from Regional Innovation Technology Development Program funded by Ministry of Land, Infrastructure and Transport of Korean government.

### REFERENCES

[ 1 ] M. -H. Chen, B. Liang, and M. Dong, "Multi-user multi-task offloading and resource allocation in mobile cloud systems," *IEEE Transaction on Wireless Communication*, vol. 17, no. 10, pp. 6790-6805, Oct. 2018

[ 2 ] A. Nadembega, A. S. Hafid, and R. Brisebois, "Mobility prediction model-based service migration procedure for follow me cloud to support QoS and QoE," in *Proceedings of the 2016 IEEE International Conference on Communications*, Kuala Lumpur, Malaysia, pp. 22-27, 2016

[ 3 ] T. Shi, W. Han, and N. Tao, "Mining aggregation moving

pattern of moving object from spatio-temporal trajectories," *Mini-micro Systems*, vol. 40, no. 5, pp. 1099-1106, 2019

[ 4 ] B. Qian, Y. Wang, R. Hong, M. Wang, and L. Shao, "Diversifying inference path selection: moving-mobile-network for landmark recognition," *IEEE Transactions on Image Processing*, vol. 30, pp. 4894-4904, May. 2021

[ 5 ] Y. -S. Lee, "Lightweight and migration optimization algorithms for reliability assurance of migration of the mobile agent," *Journal of The Korea Society of Computer and Information*, vol. 25, no. 5, pp. 91-98, May. 2020

[ 6 ] T. Thianniwet, S. Phosaard, and W. Pattara-Atikom, "Classification of road traffic congestion levels from vehicle's moving patterns: A comparison between artificial neural network and decision tree algorithm," *LNEE*, vol. 60, pp. 261-272, Feb. 2010

[ 7 ] Q. Ji and S. Zhang, "Research on sensor network optimization based on improved Apriori algorithm," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, no. 1, pp. 1-9, Nov. 2018

[ 8 ] Y. S. Lee, K. W. Nam, and M. S. Jang, "Extracting optimal moving patterns of edge devices for efficient resource placement in an FEC environment," *Journal of the Korea Institute of Information and Communication Engineering*, vol. 26, no. 1, pp. 162-169, Jan. 2022



**이연식(YonSik Lee)**

전남대학교 전자계산학과 이학사  
전남대학교 대학원 전자계산학과 이학석사  
전북대학교 대학원 전자공학과 전산응용공학  
전공 공학박사  
군산대학교 컴퓨터정보통신공학부 정교수  
※관심분야 : 에이전트 시스템, S/W 미들웨어



**장민석(MinSeok Jang)**

연세대학교 전자공학과 공학사  
연세대학교 대학원 전자공학과 공학석사  
연세대학교 대학원 전자공학과 공학박사  
군산대학교 컴퓨터정보통신공학부 정교수  
※관심분야 : IoT 시스템, 드론 응용, 딥러닝