

동적 DID Resolver 조건에서의 적응형 Resolver Fan-out 기법

유민수¹, 김진민², 김정백²
¹ 전남대학교 빅데이터융합학과
² 전남대학교 인공지능융합학과

kakaominsus@gmail.com, geonminkim@jnu.ac.kr, kyungbaekkim@jnu.ac.kr

Adaptive Resolver Fan-out for DID Resolution under Dynamic Resolver Conditions

Minsu Yoo¹, Geonmin Kim², Kyungbaek Kim²

¹Dept. of Big Data Convergence, Chonnam National University

²Dept. of Artificial Intelligence Convergence, Chonnam National University

요 약

DID resolution 을 복수 resolver 로 이중화할 때 고정 fan-out 은 resolver 상태와 무관하게 항상 같은 중복 호출 비용을 지불한다. 본 논문은 요청 이전에 관측된 이력만으로 resolver 부분집합의 성공 확률을 추정하고, 추정 성공확률이 목표를 만족하는 최소 비용 부분집합을 선택하여 first-acceptable 방식으로 실행하는 적응형 resolver fan-out 기법을 제안한다. 평가는 provider 라벨을 seed 기반 순열로 배정한 통제된 국소 환경에서 추정기 · 정책 · 비교 대상을 동결한 뒤 독립된 데이터로 1 회 수행하였다(2,640 논리 요청). 제안 기법은 사전 동결된 고정 fan-out(k=2) 기준선의 성공률 83.45%, 요청당 호출 수 2.000 에 대하여 성공률 86.70%, 요청당 호출 수 1.905 를 기록하였다. 이는 성공률 +3.26%p, 호출 4.73% 감소이며 군집 부트스트랩 95% 구간에서도 두 차이는 0 을 포함하지 않았다. 따라서 통제된 국소 환경의 독립 확인 평가에서 제안 기법은 고정 fan-out 대비 성공률을 높이는 동시에 평균 호출 수를 줄였다.

1. 서론

DID(Decentralized Identifier)는 중앙화된 제공자에 종속되지 않는 분산 식별자이며, DID resolution 은 DID 로부터 DID Document 와 메타데이터를 얻는 과정이다[1][2]. W3C DID Resolution v1 은 원격 바인딩에서 클라이언트가 복수 resolver 를 질의하여 결과를 비교할 수 있음을 명시하며[2], DIF Universal Resolver 는 여러 DID method 를 공통 구조에서 지원한다[3]. 본 연구는 표준 semantics 를 바꾸지 않고 복수 resolver 중 호출할 resolver 의 수를 결정하는 상위 orchestration 문제를 다룬다.

분산 시스템에서는 복수 replica 에 요청을 발행하고 먼저 완료된 결과를 사용하는 hedged 및 redundant request 가 tail latency 를 완화하는 방법으로 연구되어 왔다[4][5]. 최근에는 복제 서비스의 replica 선택과 scheduling 을 통해 tail latency 를 줄이는 접근도 연구되고 있다[8]. 한편 DID 관련 연구는 분산 신원관리 구

조와 DID resolution 구현 및 운영 특성을 다루고 있다[6][7]. 다만 이중화 정책은 대체로 고정적이어서, 단일 resolver 호출 시 해당 경로의 지연·불가용으로 실패할 수 있으며 복수 resolver 를 호출할 경우 상태가 양호한 구간에서도 같은 중복 비용을 지불한다.

본 연구의 기여는 (1) 관측 이력 기반 부분집합 성공 확률 추정과 최소 비용 선택으로 구성된 적응형 fan-out 구조를 제안하고, (2) 정책을 동결한 독립 확인 평가에서 고정 fan-out 대비 성공률 향상과 호출 비용 감소를 확인한 데 있다.

2. 제안 기법

2.1 문제 정의 및 부분집합 성공확률 추정

후보 resolver 집합을 $G = \{g_1, \dots, g_M\}$, 요청 시점에 정책이 관측할 수 있는 정보를 x 로 정의한다. 부분집합 $S \subseteq G(x)$ 에 대해 $q(S|x)$ 는 마감시간 τ 이내에 S 의 구성원 중 적어도 하나가 수용 가능한 결과를 반환할

확률의 추정값이며, 요청 이전에 정책이 볼 수 있는 관측만으로 계산된다. 이 추정값은 식 (1)로 계산한다.

$$\hat{q}(S|x) = \frac{\sum_{t \in H_S(x)} a_t(S) + w_0 p_0}{|H_S(x)| + w_0} \quad (1)$$

동결된 추정기는 학습 모델이 아니라 최근 관측 이력에 기반한 경험적 추정기이다. 부분집합 S 의 관측 이력 $H_S(x)$ 를 최근 $W = 10$ 개로 제한하고 식 (1)과 같이 사전 확률 방향으로 평활한 비율을 사용한다.

$a_t(S)$ 는 시점 t 에서 관측된 S 의 결과이다. $W = 10$ 과 $w_0 = 2.0$ 는 고정된 구성 상수이고, $p_0 = \frac{2196}{3192} \approx 0.688$ 는 정책 설계·동결 단계 학습 분할의 부분집합 성공 기저 비율로서 이론적 최적 확률이 아니며, 세 값은 모두 확인 평가 이전에 동결하였다. 평활 항이 없으면 초기 소수의 관측만으로 추정값이 0 또는 1로 급변한다. 관측 결과는 세 가지로 구분하며, 관측된 구성원이 마감시간 내에 수용 가능한 결과를 반환하면 1, 전원이 관측되었고 어느 것도 반환하지 못하면 0, 그 밖에는 미확정으로 두고 기록하지 않는다. 미확정을 실패로 기록하면 호출하지 않은 부분집합을 낮게 평가하는 편향이 생긴다.

x 는 요청 시점의 정책 가시 문맥이며, 동결 추정기는 학습된 feature vector가 아니라 부분집합별 관측 이력 $H_S(x)$ 를 사용한다. 평가자 전용 정보는 포함하지 않는다(3.2 절). 또한 추정은 개별 확률로부터 $1 - \prod_i (1 - p_i)$ 형태로 합성하지 않고 부분집합 단위로 수행하므로 resolver 간 독립성을 요구하지 않는다.

2.2 적응형 Resolver 부분집합 선택 및 실행

비용을 호출 수 $C(S) = |S|$ 로 두면 선택 문제는 식 (2)와 같다.

$$\begin{aligned} S^*(x) &= \arg \min_{S \subseteq \mathcal{G}(x)} C(S) \\ \text{s.t. } \hat{q}(S|x) &\geq 1 - \varepsilon \end{aligned} \quad (2)$$

$1 - \varepsilon = 0.90$ 은 trade-off를 관찰하기 위한 사전 정의

운영점이며 표준 요구값이나 이론적 최적값이 아니다. 이는 추정 성공확률에 대한 목표이므로 관측 성공률을 보장하지 않는다. 확인 평가에서 전체 동시 호출의 평균 성공률이 0.8913이었으므로 $1 - \varepsilon = 0.90$ 은 본 환경에서 비교적 높은 추정 운영 목표였다. 다만 전체 평균만으로 개별 요청 문맥의 조건부 달성 가능성을 판단할 수는 없다.

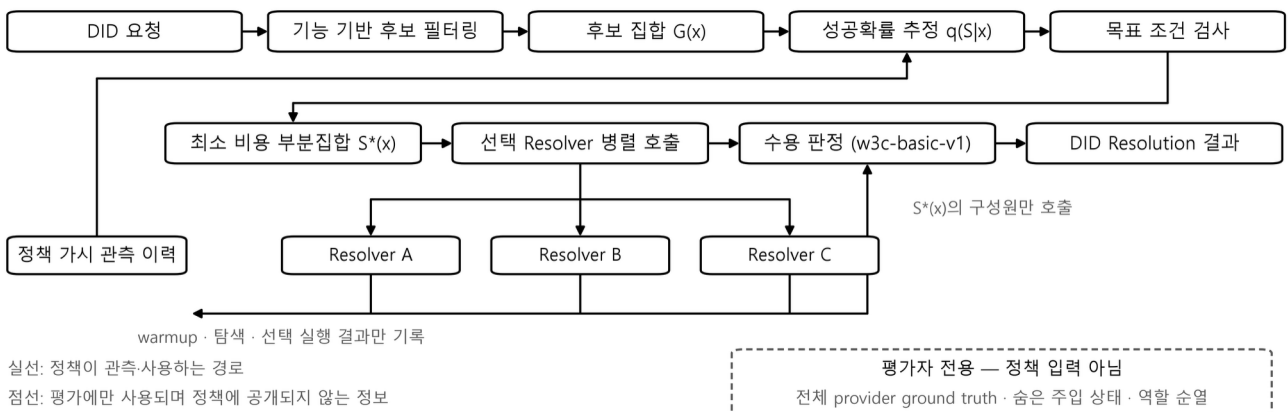
알고리즘 1. 요청별 적응형 Resolver 부분집합 선택 및 실행

Input: 후보 집합 $\mathcal{G}(x)$, 정책 가시 이력 H , 목표 $\eta = 1 - \varepsilon$, 마감 τ , 비용 $C(S) = |S|$
Output: 수용 판정을 통과한 최초 DID
Resolution 응답

- 1: $\mathcal{C} \leftarrow \{S \subseteq \mathcal{G}(x) : S \neq \emptyset\}$
- 2: for each $S \in \mathcal{C}$ do $q[S] \leftarrow \hat{q}(S|x)$ end for
- 3: $F \leftarrow \{S \in \mathcal{C} : q[S] \geq \eta\}$
- 4: if $F \neq \emptyset$ then
- 5: $S^* \leftarrow \arg \min_{S \in F} (C(S), -q[S], S)$
- 6: else
- 7: $S^* \leftarrow \arg \max_{S \in \mathcal{C}} (q[S], -C(S))$
 (목표 미충족 상태를 함께 기록)
- 8: end if
- 9: S^* 의 구성원만 동시에 호출한다
- 10: 도착 순서로 응답을 검사하여 구조적 수용 판정을 통과한 최초 응답을 return
- 11: 수용 응답이 없으면 실패를 return

알고리즘 1은 요청 한 건의 결정 논리이다. 다만 수치적 추정치만으로 해당 부분집합의 실제 관측 근거가 확보되었다고 볼 수는 없다.

알고리즘 1의 8행은 전체 호출로 확대하지 않으며 해당 요청을 목표 미충족으로 기록한다. 확인 평가 2,640건은 목표 만족 1,536건(58.2%), best-effort 744건(28.2%), 초기 warmup 360건(13.6%)으로 나뉘었고, 분기별 성공률과 요청당 호출 수는 각각 0.9128과 1.516, 0.7567과 2.179, 0.9000과 3.000이었다(사후 기술 통계).



(그림 1) 적응형 DID Resolver fan-out의 부분집합 선택 및 실행 구조

11 행의 수용 프로파일 w3c-basic-v1 은 전송 · 파싱 성공, resolution 오류 부재, DID Document 존재, DID 식별자 일치만 검사하므로 암호학적 증명 유효성이나 원장의 확정성 · 최신성을 주장하지 않는다. 또한 응답을 결합하지 않으므로 수용 판정에 실패한 응답은 도착 순서와 관계없이 최종 응답으로 채택되지 않는다.

3. 실험 설계

3.1 연구 단계와 동결 절차

본 연구는 (1) 문제 식별, (2) resolver 관측 특성화, (3) 정책 및 비교 대상의 선정 · 동결, (4) 독립 확인 평가의 네 단계로 진행하였다. 앞의 두 단계는 준비 단계이며 최종 성능 추정에 포함하지 않았다. 정책과 비교 대상은 세 번째 단계(내부 기록 V3)에서 확정된 뒤 변경하지 않았고, 최종 성능은 독립적인 네 번째 단계(내부 기록 V4)에서 평가하였다.

동결 단계에서는 생성기의 역할 대칭성을 검증하고 추정기 · 운영 규칙 · 주 비교 대상을 선정하여 동결하였다. 확인 평가에서는 추가적인 선택 · 조정 · 재학습 없이 동결 단계와 중복되지 않는 seed 를 사용하여 120 에피소드 $\times$ 22 시행 = 2,640 논리 요청을 1 회 평가하였으며, 두 단계 결과는 합산하지 않는다.

3.2 통제 Resolver 환경 및 부분 피드백

평가는 단일 호스트의 $M = 3$ 개 국소 mock resolver 로 구성된 통제 환경에서 수행하였다. 문서는 합성 문서이며 지연 · 오류 · 구조적 비수용 응답은 통제 주입 조건으로 발생시켰다. 주입 조건은 단일 provider 열화, 불안정한 응답, 구조적 비수용 위험, 두 provider 의 상관 열화, 전체 공유 열화를 포함한다. 주입 상태는 provider 이름 대신 역할로 정의한다. 에피소드마다 seed 기반 순열로 역할에 provider 라벨을 배정하므로 특정 식별자에 고정된 비대칭이 제거된다.

정책이 참조하는 이력은 부분 피드백 조건에서 수집된다. warmup, 주기적 탐색, 선택 실행의 결과만 정책에 제공되며 선택되지 않은 provider 의 동시점 응답, 숨은 주입 상태, 역할 순열은 정책 입력이 아니다. 전체 ground truth 는 평가자 전용으로 분리되어 진단과 정적 정책 재구성에만 사용되며 적응형 정책은 이를 사용하지 않는다. 에피소드마다 초기 3 회 warmup 과 주기 8 의 탐색 호출이 수행되며 모두 적응형 비용에 산입한다. 동결 규칙은 $\tau = 250$ ms, 목표 $1 - \epsilon = 0.90$ 이다.

3.3 비교 정책 및 평가 지표

표 1 은 정책별 선택 규칙과 호출 수를 제시한다.

<표 1> 비교 정책의 resolver 선택 규칙과 요청별 fan-out

정책	Resolver 선택 규칙	요청별 fan-out
최적 단일	평가자 기준으로 가장 좋은 단일 resolver	1
고정 $k = 2$	사전 동결된 두 resolver {b, c}	2
적응형	추정 성공확률이 $\hat{q}(S x) \geq 0.90$ 를 만족하는 최소 비용 부분집합	1-3
전체 동시 호출	모든 resolver	3

최적 단일은 평가자 ground truth 로부터 사후에 선정된 참조점이므로 배포 가능한 정책이 아니다. 주 비교 대상인 고정 $k = 2$ 조합은 설계 단계 학습 · 검증 분할 748 시행에서 세 후보의 정적 성공률을 동일한 기준으로 비교하여 선정하였다. 가장 높은 {local-b, local-c} (0.7968; {a,b} 0.7941, {a,c} 0.7901) 를 확인 평가 이전에 동결하였다. 세 후보의 비용은 모두 2 로 동일하여 선택에 관여하지 않으며, 확인 평가 데이터는 사용하지 않았다. 전체 동시 호출은 중복도 상한을 보이는 부차 참조점이다.

평가 지표는 모든 논리 요청을 분모로 하는 성공률 $\hat{P}_{\text{success}} = \frac{N_{\text{success}}}{N}$ 와 요청당 평균 호출 수 $\bar{C} = \frac{N_{\text{calls}}}{N}$ 이다. 성공률 차이는 대응 결과로부터 $\Delta_{\text{success}} = \frac{n_{10} - n_{01}}{N}$ 로, 호출 비용의 상대 감소는 $R_{\text{calls}} = \frac{\bar{C}_{\text{fixed}} - \bar{C}_{\text{adaptive}}}{\bar{C}_{\text{fixed}}}$ 로 정의한다.

4. 실험 결과

4.1 성공률 및 호출 비용

<표 2> 독립 확인 평가의 정책별 성능 ($N = 2,640$)

정책	성공률	요청당 호출 수
최적 단일 Resolver	0.6640	1.000
고정 $k = 2$ {b, c} (동결)	0.8345	2.000
적응형	0.8670	1.905
전체 동시 호출	0.8913	3.000

표 2 는 정책별 성능이다. 제안 기법의 성공률은 동결된 고정 $k = 2$ 대비 0.8345 에서 0.8670 으로 +3.26%p 상승하고 요청당 호출 수는 2.000 에서 1.905 로 4.73% 감소하였다(성공 86 회 증가, 호출 250 회 감소). +3.26%p 는 성공률의 절대 백분율점 증가이고 4.73%는 호출 수의 상대 감소로 단위가 다르다. 전체 동시 호출 대비로는 성공률이 약 2.42%p 낮은 대신 호출을 약 36.49% 절감한다.

4.2 대응 비교와 불확실성

<표 3> 적응형 정책과 고정 $k = 2$ 정책의 대응 성공 결과

	고정 성공	고정 실패
적응형 성공	2,150	139
적응형 실패	53	298

표 3 은 동일한 논리 요청에 대한 두 정책의 대응 결과이다. 따라서 대응 비교를 통해 요청 구성 차이에 따른 변동을 통제할 수 있으며, 순 성공률 우위는 $A_{\text{success}} = \frac{139-53}{2640} = 0.032576$ 이다.

통제 생성기는 논리 요청마다 세 provider 를 공통 시작 시점에서 동시에 조회하여 잠재 응답을 모두 기록한 뒤 정책을 재생하므로, 각 provider 의 결과는 정책의 호출 여부와 독립적으로 정의된다. 동일 요청의 ground truth 에서 재구성하였고 2,640 건 전부에서 기록 값과 일치하였다. 반면 warmup 과 탐색 호출은 적응형 비용에 포함하였으므로 보고된 호출 수는 이 관측 비용을 제외하지 않은 값이다.

불확실성은 에피소드를 재표본 단위로 하는 대응 부트스트랩으로 산출하였다. 에피소드 내 22 시행은 숨은 상태와 역할 순열을 공유하므로 독립 표본이 아니다. 따라서 120 개 에피소드를 복원추출하되 한 에피소드의 22 개 시행을 함께 이동시키고 재표본마다 정책 간 차이를 다시 계산하는 절차를 10,000 회 반복한 뒤 2.5 및 97.5 백분위수를 95% 구간으로 취하였다.

그 결과 95% 구간은 $A_{\text{success}} \in [1.21, 5.49] \%p$, $A_{\text{calls}} \in [-0.172, -0.016]$ 로 두 축 모두 유리한 방향에서 0 을 포함하지 않았다. 부트스트랩 재표본 비율은 p -값이 아니다. 다만 성공률 차이의 점추정치는 확인 평가 이전에 사전 설정한 2%p 실질성 기준을 넘지만 구간 하한(1.21%p)은 넘지 못하므로, 재현된 개선으로 해석하되 크기가 2%p 를 안정적으로 상회한다고 해석하지 않는다.

5. 결론 및 한계

본 논문은 요청 이전 관측 이력만으로 resolver 부분집합의 성공확률을 추정하고, 목표를 만족하는 최소 비용 부분집합을 first-acceptable 방식으로 실행하는 적응형 fan-out 기법을 제안하였다. 통제된 국소 환경의 독립 확인 평가에서 동결된 적응형 정책은 사전 동결된 고정 기준선보다 높은 성공률과 낮은 평균 호출 수를 보였다.

한계로는 단일 호스트의 통제 환경, 합성 문서, 주입된 지연·오류·상관 열화, 단일 마감시간·목표 수준 사용이 있다. 따라서 실제 공개 DID resolver 환경의 성능·가용성과 cold-start 경계 조건에서의 대응은 확인

되지 않았다. 또한 적응형만 페루프로 실행되었고 정적 기준선은 반사실적으로 재구성되어 관측 비용 부담에 비대칭이 있으며, 모든 provider 의 잠재 응답을 동시 조회로 기록하여 호출 감소에 따른 부하 이차 효과는 반영하지 않았다.

부분집합 단위 추정은 resolver 간 독립성 가정을 피하지만 후보 수 M 에 따라 최대 $2^M - 1$ 개의 부분집합을 고려해야 한다. 본 실험에서는 $M = 3$ 으로 7개였으며, 대규모 후보 집합과 실제 provider 환경으로의 확장이 향후 과제이다.

Acknowledgment

본 연구는 한국인터넷진흥원(KISA)-정보보안 특성화대학 지원사업의 지원을 받아 수행된 연구임(34%). 본 결과물은 농림축산식품부의 재원으로 농림식품기술기획평가원의 농식품과학기술융합형연구인력양성사업의 지원을 받아 연구되었음(RS-2024-00397026, 33%). 이 논문은 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원-지역지능화혁신인재양성사업의 지원을 받아 수행된 연구임(IITP-2026-RS-2022-00156287, 33%).

참고 문헌

- [1] M. Sporny, A. Guy, M. Sabadello, D. Reed, "Decentralized Identifiers (DIDs) v1.0," W3C Recommendation, 19 July 2022.
- [2] S. Curran, J. Andrieu (eds.), "Decentralized Identifier Resolution (DID Resolution) v1," W3C Candidate Recommendation Snapshot, 6 August 2026.
- [3] Decentralized Identity Foundation, "Universal Resolver," GitHub repository, accessed September 2026.
- [4] J. Dean, L. A. Barroso, "The Tail at Scale," Communications of the ACM, Vol.56, No.2, pp.74-80, 2013.
- [5] A. Vulimiri, P. B. Godfrey, R. Mittal, J. Sherry, S. Ratnasamy, S. Shenker, "Low Latency via Redundancy," Proc. 9th ACM CoNEXT, Santa Barbara, CA, USA, pp.283-294, 2013.
- [6] J. Xian, L. You, Q. Yi, J. Wang, G. Hu, "A survey on decentralized identity management systems," Computer Science Review, Vol.58, Art. 100811, 2025.
- [7] B. A. Alzahrani, "DID-TUF: Secure decentralized identifier management using trustless registries," Computers & Security, Vol.162, Art. 104780, 2026.
- [8] S. Ben Mokhtar, L.-C. Canon, A. Dugois, L. Marchal, E. Riviere, "A scheduling framework for distributed key-value stores and its application to tail latency minimization," Journal of Scheduling, Vol.27, No.2, pp.183-202, 2024.