기술 백서: Preference-based Multi-Objective Reinforcement Learning (Pb-MORL)

〔Paper Title / Core Technology〕

Preference-based Multi-Objective Reinforcement Learning

1. 설계 철학 및 문제 정의 (Architectural Philosophy)

기존 기술의 임계점 (Legacy Bottleneck):

• 근본 원인 (Root Cause): 기존 Multi-Objective RL (MORL)은 각 목표(Objective)들에 대한 스칼라 보상 함수(Scalar Reward Function)가 사전에 완벽하게 정의되어 있다고 가정함.

• 치명적 한계 (Critical Failure): 현실 세계의 복잡한 다목적 문제(예: 자율주행의 안전 vs 승차감 vs 속도)를 단순한 수치적 보상으로 표현하는 것은 불가능에 가까움. 잘못 설계된 보상 함수는 Reward Hacking이나 Oversimplification을 초래하여, 실제 의도와 다른 괴상한 정책을 유도함.

• 패러다임 전환 (Paradigm Shift): 본 논문은 "보상 함수 설계"라는 난제를 우회하기 위해, 인간의 선호(Preference) 데이터를 직접 활용하여 명시적인 다목적 보상 모델(Explicit Multi-Objective Reward Model)을 학습하는 Pb-MORL 프레임워크를 제안함. 이를 통해 복잡한 보상 설계 없이도 파레토 최적(Pareto Optimal) 정책을 학습할 수 있음을 증명함.

개념 시각화 (Conceptual Analogy):

〉〔Analogy〕 요리 대회에서 심사위원이 "소금 3g, 설탕 5g"처럼 정확한 점수 기준(Reward Function)을 주는 것이 아니라, 두 요리를 먹어보고 "A가 B보다 낫다"는 선호(Preference)만 표현하면, 요리사(Agent)가 그 선호 패턴을 학습하여 최고의 요리(Optimal Policy)를 만들어내는 것과 같음.

2. 수학적 원리 및 분류 (Mathematical Formalism)

시스템 분류 (System Taxonomy):

• 아키텍처 유형: Preference-based Multi-Objective Reinforcement Learning (Pb-MORL)

• 알고리즘 기반: Envelope Q-Learning (EQL) + Bradley-Terry Preference Model

• 불변 특성: Pareto Optimality Guarantee (Under Condition of Consistent Preferences)

핵심 수식 및 상세 해설 (Core Formulation & Breakdown):

1. Bradley-Terry Preference Model:

P_ψ〔σ ₁ ≻ σ ₂ | w〕= fracexp(Σ γ t w r_ψ (s₁, a₁)) Σ_i ∈ 0, 1 exp(Σ γ t w r_ψ (s, a))

• Variable Definition (변수 정의):

• P_ψ : 학습된 보상 모델 r_ψ 를 기반으로 한 선호 확률 예측기

• σ , σ ₁: 비교 대상이 되는 두 궤적(Trajectory Segment)

• w: 선호 판단의 기준이 되는 가중치 벡터 (Weight Vector). w ∈ mathbbRm, sum w = 1.

• r_ψ (s, a): 파라미터 ψ 로 학습되는 명시적 다목적 보상 모델 (Vector)

• ≻: 선호 관계 (Preferred over)

• Physical Meaning (수식의 물리적 의미):

• 두 궤적 간의 선호 확률은 각 궤적의 누적 보상(학습된 보상 모델 값)의 지수 함수적 비율로 결정된다. 즉, 학습된 보상 모델 r_ψ 의 합이 클수록 더 선호될 확률이 높게 모델링된다. 이는 선호 데이터를 통해 역으로 보상 함수를 추정할 수 있게 해주는 핵심 연결 고리이다.

2. Cross-Entropy Loss (Reward Learning):

L(ψ) = - mathbbE_(σ , σ ₁, w, p) sim D ≤ft〔 p log P_ψ〔σ ₁ ≻ σ ₂ | w〕+ (1-p) log P_ψ〔σ ₂ ≻ σ ₁ | w〕right〕

• Variable Definition:

• p: 실제 인간(또는 교사)의 선호 레이블 (0, 0.5, 1)

• D: 수집된 선호 데이터셋

• Physical Meaning:

• 예측된 선호 확률 P_ψ 와 실제 선호 p 사이의 차이(엔트로피)를 최소화한다. 이를 통해 보상 모델 r_ψ 는 인간의 선호 판단 기준을 모사하게 된다.

3. Multi-Objective Q-Learning (Policy Optimization):

J(π) = mathbbE_w sim Dw ≤ft〔 w Q(s, a, w) right〕

• Physical Meaning:

• 학습된 보상 모델 r_ψ 를 통해 계산된 Q 값을 최대화하는 방향으로 정책을 업데이트한다. 여기서 Q 값은 다목적 벡터이며, 가중치 w와의 내적을 통해 스칼라화된 효용(Utility)을 극대화한다.

3. 실행 파이프라인 및 데이터 흐름 (Execution Pipeline)

입력 명세 (Trace Spec):

• Input Context: State Vector `〔Batch, State_Dim〕`, Weight Vector `〔Batch, Objective_Dim〕`

• 예시: `s=〔10.5, 3.2〕`, `w=〔0.7, 0.3〕` (속도 중시형)

순전파 로직 (Forward Propagation Logic):

1. Preference Query & Collection:

• Action: 에이전트가 환경과 상호작용하여 궤적 쌍 (σ , σ) 생성.

• Feedback: 교사(Teacher)가 가중치 w 하에서 선호 p를 제공.

• Data Storage: `Buffer` ≤ftarrow `(\sigma_0, \sigma_1, w, p)`

2. Reward Model Forward:

• Input: Segment σ (State-Action Sequence)

• Process: Neural Network ψ 가 각 (s,a)에 대해 벡터 보상 r_ψ (s,a) 출력.

• Aggregation: sum γ t w r_ψ (s,a) 계산하여 Segment Utility 산출.

• Output: Preference Probability P_ψ

3. Policy Forward (EQL):

• Input: State s, Weight w

• Process: Q-Network θ 가 다목적 Q-Value Q(s, a, w) 예측.

• Selection: argmax w Q(s, a, w)를 통해 행동 선택.

4. 학습 메커니즘 및 최적화 (Optimization Dynamics)

역전파 역학 (Backpropagation Dynamics):

• Step 1: Reward Model Update (보상 학습)

• Principle: Cross-Entropy Loss 최소화.

• Purpose: 보상 모델 r_ψ 가 인간의 선호 판단과 일치하도록 조정.

• Data Example: 실제 선호가 σ 인데 모델이 σ 를 선호한다고 예측(확률 0.8)했다면, 큰 Loss 발생 rightarrow σ 의 보상을 높이고 σ 의 보상을 낮추는 방향으로 Gradient 발생.

• Step 2: Relabeling (데이터 갱신)

• Principle: Off-policy Learning을 위한 데이터 재가공.

• Purpose: 과거에 수집된 궤적들의 보상 값을 현재 시점의 학습된 보상 모델 r_ψ 값으로 덮어씌움(Relabeling). 이를 통해 정책 학습이 최신 보상 모델을 반영하도록 함.

• Step 3: Policy Update (정책 최적화 - EQL)

• Principle: Bellman Error Minimization.

• Equation: L(θ) = | y - Q(s, a, w) |², where y = r_ψ + γ max_a′ Q(s′, a′, w).

• Purpose: 학습된(가상의) 보상 r_ψ 를 최대화하는 행동을 학습.

알고리즘 구현 (Pseudocode Strategy on Pythonic Logic):

```
def train_pbmorl():
    # 1. Initialize Networks
    reward_model = RewardNet()
    policy = EQLAgent()
    buffer = PreferenceBuffer()

    for iteration in range(Max_Iter):
        # 2. Collect Data & Query Preference
        segments = collect_segments(policy)
        preferences = teacher.query(segments) # Human or Scripted feedback
        buffer.add(segments, preferences)

        # 3. Update Reward Model
        for _ in range(Grad_Steps):
            loss_r = compute_cross_entropy(reward_model, buffer.sample())
            optimizer_r.step(loss_r)

        # 4. Relabel & Update Policy
        replay_buffer.relabel_rewards(reward_model)
        for _ in range(Policy_Steps):
            loss_q = compute_bellman_error(policy, replay_buffer.sample())
            optimizer_q.step(loss_q)
```

5. 구현 상세 및 제약 사항 (Details & Constraints)

안정화 기법 (Stabilization Techniques):

• Critical Component: Weight Conditioned Network와 Relabeling.

• Justification: 가중치 w가 바뀔 때마다 별도의 모델을 학습하는 것은 비효율적임. w를 입력으로 받는 하나의 네트워크로 모든 선호 분포를 커버함. 또한, 보상 모델이 변하기 때문에 과거 데이터의 보상 값을 주기적으로 갱신(Relabeling)하지 않으면 정책 학습이 수렴하지 않음.

시스템 한계 (System Limitations):

• Computational Complexity: 보상 모델 학습과 정책 학습이 번갈아 진행되므로, 단일 RL 대비 학습 시간이 증가함.

• Resource Constraints: 대량의 선호 데이터(Query)가 필요할 수 있음. 인간 피드백 비용이 높은 경우 병목이 될 수 있음.

6. 산업 적용 전략 (Industrial Application)

비즈니스 가치 (Business Value Proposition):

• Operational Efficiency: 복잡한 공정 제어(예: 반도체 제조, 화학 플랜트)에서 전문가가 일일이 보상 함수를 코딩할 필요 없이, "이 결과가 저것보다 좋다"는 피드백만으로 제어기를 최적화할 수 있음.

• Use Case:

• 스마트 그리드: 에너지 효율 vs 안정성 vs 비용 균형 조절.

• 헬스케어: 환자의 고통 최소화 vs 치료 효과 극대화 (정량화하기 힘든 지표).

• 개인화 추천: 사용자마다 다른 선호 가중치(w)를 실시간으로 반영하여 추천 로직 변경.

7. 검증 및 누락 점검 (Validation Agent - Self-Correction)

Missing Information Check:

- 〔x〕 Core Equations: Bradley-Terry Model, Cross-Entropy Loss, EQL Objective 포함됨.
- 〔x〕 Key Algorithms: Reward Learning rightarrow Relabeling rightarrow Policy Update 루프 포함됨.
- 〔x〕 Theorems: 본문에는 자세한 증명 과정생략되었으나, "Theorem 4 (보상 모델 최적화와 파레토 최적화의 동치성)"의 의미는 1절과 2절에 반영됨.

Final Polish:

- 기술적 인과관계("Why & How")를 중심으로 서술되었으며, 모호한 표현을 배제함.