

# Task Planning과 Job Scheduling 기술 동향: 전통적 기법에서 AI 기반 접근까지

## Task Planning and Job Scheduling: From Traditional Methods to AI-based Approaches

조준면 (J.M. Cho, jmcho@etri.re.kr)

필드로보틱스연구실 책임연구원

### ABSTRACT

Task planning and job scheduling are fundamental components of intelligent systems, focusing on “what to do” and “when to execute.” Although they have been studied separately in artificial intelligence and operations research, they are closely related in practice and often require integration. Task planning has evolved from symbolic and constraint-based approaches (e.g., STRIPS and PDDL) to learning-based and neurosymbolic methods and, more recently, to large language model (LLM)-based approaches. Job scheduling has progressed from optimization and heuristics to metaheuristics, reinforcement learning, and graph neural networks. Their convergence, known as integrated process planning and scheduling, highlights the need for joint optimization in real-world environments. Applications include smart manufacturing, cloud and edge computing, and robotics. Key challenges remain in handling uncertainty and complex constraints and in ensuring explainability. Future research directions include neurosymbolic integration, GNN- and DRL-based optimization, and LLM-assisted modeling.

**KEYWORDS** AI Planning, Graph Neural Networks, Integrated Process Planning and Scheduling, Job Scheduling, 작업 계획, 작업 스케줄링

## I. 서론

4차 산업혁명과 함께 지능형 제조, 자율 시스템, 클라우드 및 엣지 컴퓨팅 환경이 확산되면서 복잡한 작업을 효율적으로 계획하고 실행하는 기술의 중요성이 점차 커지고 있다. 이러한 맥락에서 Task

Planning과 Job Scheduling은 지능형 시스템의 두 핵심 축으로 자리매김하고 있다. Task Planning은 주어진 목표를 달성하기 위한 일련의 행위(Action)들을 순차적으로 구성하는 문제이며, Job Scheduling은 주어진 자원과 제약 조건하에서 작업(Job)을 최적으로 배치하고 실행 시점을 정하는 문제다. 두 분야는

\* DOI: <https://doi.org/10.22648/ETRI.2025.J.400604>

\* This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) [No. 2022-0-00187 and No. 2022-0-01049].

연구 배경과 전통적 접근 방식에서 차이를 보이지만, 실제 산업 환경에서는 밀접하게 연관되어 있으며 상호 보완적 역할을 수행한다[1,6].

Task Planning은 인공지능(AI) 분야에서 주로 다루어져 왔으며, 대표적으로 STRIPS, PDDL 기반의 기호적 계획(Symbolic Planning)에서 출발하여 제약 기반 계획(CBP: Constraint-Based Planning), 확률적 · 시간적 계획(Probabilistic/Temporal Planning) 등으로 발전하였다[1]. 반면, Job Scheduling은 전통적으로 운영과학(OR: Operations Research)과 산업공학 분야에서 연구되었으며, 혼합정수계획법(MILP), 제약 충족 문제(CSP), 휴리스틱 및 메타휴리스틱을 중심으로 이론적 · 실무적 발전을 거듭해 왔다[7,8]. 최근에는 두 분야 모두에서 기계학습(ML)과 심층강화학습(DRL), 그리고 그래프 신경망(GNN) 등 데이터 기반 접근이 활발히 시도되고 있으며, 더 나아가 생성형 대규모 언어모델(LLM)을 접목한 새로운 연구가 빠르게 등장하고 있다[1,2,9-12].

특히 스마트 팩토리, 클라우드 워크플로우, 로봇틱스, 자율주행 등 실제 응용 분야에서는 계획과 스케줄링이 동시에 요구된다. 예를 들어, 로봇은 주어진 목표를 달성하기 위해 작업 순서를 계획(Task Planning)해야 하며, 동시에 다수의 로봇이나 기계가 공유 자원을 효율적으로 사용하도록 작업 실행 순서를 최적화(Job Scheduling)해야 한다. 따라서 두 기술을 통합적으로 고려하는 연구, 즉 Planning-Scheduling Integration이 새로운 연구 패러다임으로 부상하고 있다[9,13].

본고에서는 Task Planning과 Job Scheduling 기술 동향: 전통적 기법에서 AI 기반 접근까지를 주제로, 두 분야의 정의와 관계, 그리고 기술 발전 흐름을 종합적으로 살펴본다. 먼저 Task Planning과 Job Scheduling의 개념을 정리하고, 각 기술의 발전 동향을 전통적 기법에서 학습 기반 기법, 나아가 생성

형 거대모델 기반 접근까지 체계적으로 분석한다. 이어서 두 분야 간의 관계와 차이점을 논의하고, 통합적 접근 방식의 최신 연구 동향을 소개한다. 또한, 실제 산업 현장에서의 응용 사례와 적용 현황을 살펴보고, 현장에서의 도전 과제와 향후 발전 방향을 제시한다. 이를 통해 본고는 Task Planning과 Job Scheduling의 기술 발전 흐름을 총체적으로 정리하고, 차세대 지능형 시스템 구현을 위한 연구 방향을 제시하고자 한다.

## II. Task Planning 개요 및 기술 동향

### 1. Task Planning의 정의

Task Planning은 주어진 초기 상태에서 목표 상태를 달성하기 위해 수행해야 할 일련의 행위(Action)들을 순차적으로 도출하는 문제를 의미한다[1]. 전통적으로 인공지능(AI) 연구 분야에서 다루어져 왔으며, 복잡한 환경에서의 합리적 의사결정을 가능하게 하는 핵심 기술이다.

대표적으로 고전적 계획(Classical Planning)은 세계를 상태(State)의 집합으로 보고, 초기 상태에서 목표 상태로 전이하는 행위들의 시퀀스를 탐색하는 방식으로 이해된다. 이때 각 행위는 전제조건(Precondition)과 효과(Effect)가 명확히 기술되어야 하며, 이를 통해 상태 공간을 정의하고, 휴리스틱 검색 방법을 활용하여 목표 상태에 도달하는 경로를 빠르게 탐색하는 접근을 취한다.

### 2. 전통적 기법

Task Planning의 초기 연구는 기호적(Symbolic) 접근을 기반으로 발전하였다.

- **STRIPS:** 1970년대 초 등장한 대표적 기호 계획 언어로, 행위(Action)의 전제조건과 효과를

논리적으로 정의한다.

- **PDDL(Planning Domain Definition Language):** 다양한 계획 문제를 표준화된 방식으로 기술하기 위해 제안된 언어로, 이후 국제 계획 대회(IPC)의 표준 표현으로 활용되고 있다.
- **제약 기반 계획(Constraint-Based Planning):** 계획 문제를 제약 충족 문제(CSP)로 모델링하여, 제약 조건을 만족하는 실행 가능한 계획을 탐색하는 접근이다.

이러한 전통적 기법들은 문제의 명확한 모델링이 가능할 경우 최적성 보장이 가능하지만, 실제 복잡한 환경에서는 표현력 한계와 연산 복잡도 문제를 안고 있다.

### 3. 학습 기반 접근

2000년대 이후 기계학습과 강화학습 기법이 발전하면서, Task Planning에도 데이터 기반 학습 기법이 도입되었다[1,2,6].

- **강화학습(RL: Reinforcement Learning):** 환경과 상호작용하여 목표 달성 정책(Policy)을 학습한다. Tree Search, Monte Carlo Tree Search(MCTS)와 결합하여 장기적 계획 문제 해결에 활용된다.
- **심층학습(Deep Learning) 결합:** 복잡한 상태 표현을 신경망으로 추상화하고, 휴리스틱 함수 학습을 통해 탐색 효율성을 높이는 연구가 활발하다.
- **뉴로심볼릭(Neuro-Symbolic) 접근:** 신경망 기반 학습과 기호적 계획의 강점을 결합하여, 불확실성과 구조적 추론을 동시에 처리할 수 있는 방향으로 발전하고 있다.

이러한 학습 기반 접근은 대규모 데이터와 시물

표 1 전통적 Task Planning 기법과 학습 기반 기법 비교

구분	전통적 기법	학습 기반 기법
접근 방식	기호 기반 탐색, 제약 충족	데이터 기반 학습, 정책 최적화
장점	최적성 보장, 해석 용이	복잡한 환경 적응, 경험 기반 일반화
한계	모델링 비용, 연산 복잡성	설명력 부족, 학습 데이터 의존적
응용	고전적 로봇틱스 계획, 자동화 테스트	게임, 자율주행, 대규모 시뮬레이션

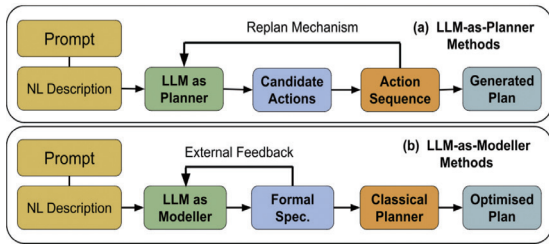
레이션을 활용할 수 있는 환경에서 강점을 보이지만, 여전히 설명 가능성 부족과 안정성 문제가 주요 한계로 지적된다. 표 1이 전통적 자동 계획 기법과 최근의 학습 기반 기법의 특징을 비교하여 보여준다.

### 4. 거대 언어모델 기반 접근

최근 ChatGPT와 같은 대규모 언어모델(LLM)의 발전은 Task Planning 연구에도 큰 변화를 불러왔다. LLM은 자연어로 기술된 목표와 환경을 해석하고, 계획 문제를 PDDL 등 형식적 표현으로 자동 변환하는 데 활용될 수 있다[3-5].

- **LLM-as-Planner:** LLM이 직접 계획을 생성하는 방식. 직관적이고 빠르지만, 장기적 일관성 부족 문제를 가진다.
- **LLM-as-Modeler:** LLM이 도메인 지식과 문제 정의를 추출하여 기호적 계획기의 입력 모델을 자동 생성한다. 신뢰성과 설명력이 더 높다.
- **LLM+Search Hybrid:** LLM이 생성한 후보 계획을 기존 탐색 기반 계획기가 검증·보완하는 방식이다.

그림 1이 두 가지 LLM 활용 방식을 도식적으로



출처 Reprinted from M. Tantakoun et al., "LLMs as Planning Modelers: A Survey for Leveraging LLMs to Construct Automated Planning Models," arXiv preprint, 2025. doi: 10.48550/arXiv.2503.18971 [4].

**그림 1 LLM-Planning 형태: (a) LLM-as-Planner, (b) LLM-as-Modeler**

보여준다. 이러한 접근은 실제 사용자 친화적 인터페이스를 제공할 수 있으며, 도메인 전문가가 아닌 일반 사용자도 계획 문제를 정의·활용할 수 있게 하는 잠재력을 가진다. 다만, 여전히 LLM의 추론 안정성, 정확성, 안전성 확보가 해결해야 할 과제이다.

### III. Job Scheduling 개요 및 기술 동향

#### 1. Job Scheduling의 정의

Job Scheduling은 제한된 자원(Resource)하에서 다수의 작업(Job)을 효율적으로 배치하고 실행 시점을 결정하는 문제로, 생산성 향상과 자원 최적화를 목적으로 한다. 대표적인 목적 함수로는 총 작업 완료 시간(Makespan) 최소화, 지연(Tardiness Minimization) 최소화, 자원 활용도 최대화 등이 있으며, 제조, 물류, 클라우드 컴퓨팅 등 다양한 산업 현장에서 핵심적인 역할을 수행한다[6-8].

Job Scheduling은 조합 최적화(Combinatorial Optimization) 문제의 대표적 예시로 알려져 있으며, 대부분 NP-hard 문제로 분류된다. 따라서 다양한 수리적, 휴리스틱, 메타휴리스틱, 학습 기반 접근이 연구되어 왔다.

#### 2. 전통적 기법

Job Scheduling의 초기 연구는 수리적 최적화와 제약 기반 기법 중심으로 전개되었다.

- **정확 해법(Exact Methods):** 혼합정수계획법(MILP: Mixed-Integer Linear Programming), 동적계획법(Dynamic Programming), 분기한계법(Branch and Bound) 등이 대표적이다[8]. 이들은 최적 해를 보장하지만, 문제 크기가 커지면 계산 복잡도가 기하급수적으로 증가한다.
- **제약 기반 스케줄링(Constraint-Based Scheduling):** 자원 제약, 선행 제약 등을 제약 충족 문제(CSP: Constraint Satisfaction Problem)로 모델링하여 효율적으로 탐색한다.
- **휴리스틱(Heuristics):** 최단 처리시간 우선(SPT: Shortest Processing Time), 최소 여유시간 우선(Least Slack Time) 등 규칙 기반 접근으로 빠른 계산이 가능하다.

이러한 기법들은 소규모 문제에서는 효과적이지만, 현실 산업 현장의 대규모·동적 환경에서는 한계가 있다.

#### 3. 메타휴리스틱 기반 접근

산업 현장에서의 복잡성과 대규모 문제를 해결하기 위해 메타휴리스틱 알고리즘이 활발히 적용되었다.

- **유전 알고리즘(GA: Genetic Algorithm):** 진화 연산을 통해 해 탐색[10,11]
- **시뮬레이티드 어닐링(SA: Simulated Annealing), 담금질 기법:** 전역 탐색에 유리
- **개미 집단 최적화(ACO: Ant Colony Optimization), 입자 군집 최적화(PSO: Particle Swarm**

**표 2 전통적 Job Scheduling 기법과 메타휴리스틱 기법 비교**

구분	전통적 기법	메타휴리스틱 기법
특징	최적성 보장	근사해 탐색
장점	정확성, 이론적 기반	대규모 문제 적용 가능, 유연성
한계	계산량 증가, 대규모 문제 부적합	최적성 보장 불가, 파라미터 민감
대표 기법	MILP, DP, CSP	GA, SA, ACO, PSO, Tabu Search

**Optimization):** 집단 지능 기반 탐색으로 다양한 Job Shop Scheduling Problem(JSSP)에 적용[8]

- **Tabu Search, Variable Neighborhood Search** 등: 지역 탐색과 탐색 공간 확장을 결합

메타휴리스틱은 근사 해법을 빠르게 제공하지만, 최적성 보장은 어렵고 파라미터 설정이 결과에 큰 영향을 준다. 표 2가 전통적 기법과 메타휴리스틱 기법의 차이를 비교하여 보여준다.

## 4. 학습 기반 접근

최근에는 데이터 기반 학습 기법이 Scheduling 문제 해결에 적극적으로 도입되고 있다[10-13].

- **강화학습(RL: Reinforcement Learning):** 스케줄링 문제를 순차적 의사결정(MDP: Markov Decision Process)으로 모델링하여, 정책(Policy)을 학습하는 방식
- **심층 강화학습(DRL: Deep Reinforcement Learning):** Deep Q-Network(DQN), Proximal Policy Optimization(PPO) 등과 결합하여 대규모 문제 해결에 활용
- **그래프 신경망(GNN: Graph Neural Network):** 스케줄링 문제를 그래프 구조(작업-자원 관계)

로 표현하여, 구조적 제약을 학습 기반으로 처리하는 방법

이러한 접근은 전통적 기법 대비 대규모, 동적 환경에 강점을 가지며, 다양한 제조/클라우드/로보틱스 응용으로 확산되고 있다. 그러나 여전히 일반화 성능, 학습 비용, 안정성 문제는 주요 과제이다.

## 5. 최근 동향: 생성형 모델 및 LLM 접목 가능성

최근 연구에서는 생성형 AI 및 대규모 언어모델(LLM)을 Scheduling 문제에 적용하려는 시도가 이루어지고 있다.

- **LLM+Scheduling Hybrid:** LLM이 문제 정의 및 제약 조건을 자동으로 기술하고, 전통적 최적화/학습 기반 기법이 실제 스케줄링을 수행하는 구조
- **설계 자동화:** 제조공정에서 작업 제약을 자연어로 입력하면, LLM이 이를 스케줄링 모델(PDDL, MILP 등)로 변환
- **설명 가능성:** 기존 강화학습 기반 스케줄링의 '블랙박스성'을 보완하는 도구로서 활용 가능성이 탐구되고 있다[3,9].

이는 아직 초기 연구 단계지만, 향후 Job Scheduling의 자동화·지능화를 가속화할 잠재력이 크다.

## IV. Task Planning과 Job Scheduling의 관계

### 1. 개념적 차이와 공통점

Task Planning과 Job Scheduling은 모두 주어진 목표를 달성하기 위한 실행 계획을 수립한다는 점에



서 공통점이 있다. 두 문제 모두 상태(State), 행위(Action) 혹은 작업(Job), 제약(Constraints)을 고려하여 최적의 실행 순서를 도출한다. 그러나 초점과 범위에 있어 중요한 차이가 존재한다[6,9].

Task Planning은 “무엇을 할 것인가(what to do)”에 초점을 둔다. 초기 상태와 목표 상태를 정의하고, 목표를 달성하기 위해 가능한 행위들의 조합을 탐색한다. 주로 행위의 논리적 전제조건과 효과를 다루며, 문제 정의가 주로 AI Planning 언어(PDDL 등)로 표현된다[1].

Job Scheduling은 “언제, 어떤 자원으로 할 것인가(when and with what resources)”에 초점을 둔다. 작업(Job) 집합, 자원(Resource), 처리시간(Processing Time), 기계(Machine) 제약 등을 고려해 시간과 자원 배분 문제를 해결한다[6,7].

따라서 Planning이 목표 달성 경로를 설계한다면, Scheduling은 해당 경로를 자원 제약하에서 실행 가능한 일정표로 구체화하는 역할을 수행한다고 볼 수 있다.

## 2. 두 분야의 연구 전통 비교

Task Planning은 주로 인공지능(AI) 커뮤니티에서 연구되었고, Job Scheduling은 운용과학(OR: Operations Research) 및 산업공학 커뮤니티에서 발전해 왔다[6].

- **AI Planning 전통:** 기호적 표현을 통한 상태-행위 기반 모델링, 탐색 알고리즘, 휴리스틱 함수 발전에 집중. 최근에는 심층학습과 LLM을 활용한 모델링 자동화로 확장되고 있다.
- **OR Scheduling 전통:** 수리적 최적화, 제약 충족, 휴리스틱 및 메타휴리스틱 기법에 기반. 다목적 최적화와 불확실성 고려 기법(Robust/Online Scheduling 등)으로 발전해 왔다.

최근에는 Planning 커뮤니티와 Scheduling 커뮤니티 간 학술대회 · 워크숍에서 통합 연구 세션이 늘어나고 있으며, Integrated Process Planning and Scheduling(IPPS) 연구가 대표적 사례다[13].

## 3. 상호 보완적 역할

현실 응용에서는 Task Planning과 Job Scheduling을 분리해서는 효과적이지 않은 경우가 많다. 예를 들어, 스마트 제조공정에서 제품 조립을 위한 작업 계획(Task Planning)을 수립하더라도, 생산 라인의 기계 가용성이나 작업자 배치와 같은 제약을 반영한 일정(Job Scheduling)이 함께 고려되어야 한다.

Planning 단계에서 산출된 작업 순서와 요구 자원 정보는 Scheduling 단계의 입력으로 활용된다. 예컨대, 로봇 작업 계획이 먼저 수립되면 이를 기반으로 실제 실행 가능한 시간표가 만들어진다.

한편, Scheduling 과정에서 발생하는 자원 부족, 지연, 장비 고장과 같은 현실적 제약 및 동적 상황은 다시 Planning 단계로 피드백되어 계획을 수정 · 보완하는 데 사용된다.

결과적으로, 현실에서는 통합적 해결의 필요성이 존재한다. 최근 연구들은 계획과 스케줄링을 동시에 고려하는 통합 최적화 접근이 현실성과 효율성 측면에서 더 효과적임을 보여주고 있다. 특히 그래프 신경망(GNN)과 강화학습(DRL)을 결합한 방법은 계획과 스케줄링을 통합적으로 최적화하는 데 강점을 보이는 것으로 보고된다[12].

따라서 Task Planning과 Job Scheduling은 각각 독립적으로 유용하지만, 실제 산업 현장에서는 상호 보완적으로 작동해야 한다. “무엇을 할 것인가(Planning)”와 “언제 · 어떻게 실행할 것인가(Scheduling)”를 함께 고려할 때, 지능형 시스템은 비로소 효율성과 현실성을 동시에 확보할 수 있다.

## V. Task Planning과 Job Scheduling의 통합적 접근

### 1. 통합 문제 정의

Task Planning과 Job Scheduling은 본질적으로 서로 다른 문제로 발전해 왔으나, 실제 산업 현장에서는 두 문제를 동시에 고려해야 하는 경우가 많다. 이러한 문제는 일반적으로 Integrated Process Planning and Scheduling(IPPS) 문제로 정의되며, 목표 달성을 위한 작업 순서를 계획하는 동시에, 자원과 시간 제약을 고려한 일정 최적화를 함께 수행한다.

통합 문제는 단일 문제로 풀 경우 계산 복잡도가 많이 증가하지만, 계획과 스케줄링 간 상호작용을 명시적으로 고려함으로써 현실적인 해법을 도출할 수 있다는 장점이 있다.

### 2. 전통적 통합 접근

초기 연구에서는 계획과 스케줄링을 순차적으로 연결하거나, 제한적으로 통합하는 방식이 주류였다.

계층적 접근(Hierarchical Approach)에서는 먼저 Planning을 수행한 뒤, 그 결과를 Scheduling에 전달하여 세부 일정을 작성하였다. 이 방식은 단순하지만 계획과 일정 간 불일치가 자주 발생한다.

제약 기반 통합(Constraint-Based Integration)에서는 계획과 일정 제약을 동시에 모델링하여 일관된 해를 탐색, 제약 충족 문제(CSP)나 수리적 최적화 모델로 IPPS 문제를 정식화한 사례가 보고되었다.

### 3. AI 기반 통합 접근

최근에는 학습 기반 AI 기법을 활용하여 IPPS 문제를 해결하려는 연구가 활발히 진행되고 있다. 강화학습 기반 통합에서는 작업 순서와 자원 배정을

동시에 학습하는 방식으로, 클라우드 및 엣지 컴퓨팅 환경의 워크플로우 최적화에 적용된 사례가 보고되었다.

그래프 신경망(GNN) 기반 통합에서는 공정 계획과 스케줄링 문제를 그래프 구조로 표현하고, 이를 학습하여 복잡한 상호 의존성을 효과적으로 처리하는 연구가 제안되었다.

한편, 뉴로심볼릭 접근에서는 기호적 계획 모델의 논리적 구조와 심층학습의 일반화 능력을 결합하여, 계획-스케줄링 문제를 통합적으로 풀려는 시도가 이어지고 있다.

## VI. 산업 응용 및 도전 과제

### 1. 주요 응용 분야

Task Planning과 Job Scheduling은 제조업, 클라우드 및 엣지 컴퓨팅, 로봇틱스와 자율 시스템 등 다양한 산업 분야에서 활발히 응용되고 있다. 제조업에서는 공정 계획(Process Planning)과 작업 스케줄링(Job Scheduling)을 통합적으로 고려하여 생산 라인을 최적화하는 연구가 꾸준히 진행되고 있으며, 특히 스마트 팩토리에서는 이를 통해 납기 단축, 비용 절감, 자원 활용 극대화를 실현하고 있다.

클라우드 환경에서는 워크플로우 스케줄링을 통해 데이터 처리 작업을 효율적으로 자원에 배치함으로써 실행 시간을 단축하고 비용을 절감할 수 있으며, 엣지 컴퓨팅에서는 IoT 기기와 엣지 서버 간의 작업 스케줄링을 통해 지연(Latency)을 최소화하고 에너지 효율을 높이는 사례가 보고되고 있다.

또한, 로봇틱스 분야에서는 물체 조작이나 경로 계획에서 고전적 계획 기법과 학습 기반 계획 기법이 모두 활용되고 있으며, 다중 로봇 시스템의 경우 개별 로봇의 Task Planning과 전체 시스템 차원의 Job Scheduling을 결합하여 협력적 작업 수행을

가능하게 한다. 더 나아가 자율주행 차량과 같은 응용에서는 주행 경로를 설계하는 Task Planning과 도로 및 교통 제약을 고려한 흐름 관리에 해당하는 Job Scheduling이 함께 요구되며, 이를 통해 안정적이고 효율적인 자율주행이 구현된다.

## 2. 산업 적용 시 도전 과제

산업 현장에서 Task Planning과 Job Scheduling 기술을 적용하는 과정에는 여러 현실적 제약이 존재한다. 우선, 실제 환경은 장비 고장, 긴급 주문, 자원 가용성 변동과 같은 불확실성과 동적 요소가 빈번하게 발생한다. 그러나 현재의 계획 및 스케줄링 시스템은 이러한 변화를 실시간으로 반영하는 데 한계가 있으며, 결과적으로 계획과 일정의 안정성이 떨어지는 문제가 나타난다.

또한, 현대 제조 환경이 다품종 소량 생산이나 맞춤형 주문으로 변화하면서 고려해야 할 제약 조건은 점점 복잡해지고 있다. 전통적인 최적화 기반 접근은 이러한 복잡한 제약을 모두 반영하기 어렵고, 문제의 크기가 커질수록 계산 비용이 급격히 증가하여 현실적인 의사결정 지원에 제약을 준다.

학습 기반 기법의 확산에도 불구하고 데이터와 모델링 비용 역시 중요한 도전 과제로 남아 있다. 강화학습이나 딥러닝 기반 접근은 대규모 학습 데이터와 시뮬레이션 환경을 필요로 하는데, 실제 산업 현장에서 이를 구축하는 데는 상당한 시간과 비용이 소요된다.

마지막으로, 설명 가능성과 신뢰성 부족도 산업 적용의 큰 걸림돌이다. 특히 강화학습이나 딥러닝 기반 스케줄링 기법은 ‘블랙박스’ 특성을 보여 결과의 근거를 명확히 설명하기 어렵다. 이는 산업 현장에서 요구되는 안정성, 안전성, 규제 준수 측면에서 한계를 드러내며, 결과적으로 기술 도입 속도를 늦

추는 요인으로 작용하고 있다.

## VII. 향후 기술 발전 방향

Task Planning과 Job Scheduling은 이미 다양한 산업 분야에서 핵심적 역할을 하고 있지만, 여전히 해결해야 할 과제가 많다. 향후 기술 발전은 전통적 기법과 최신 AI 기법을 융합하면서, 실제 산업 환경에서 요구되는 실시간성, 안정성, 설명 가능성을 확보하는 방향으로 전개될 것으로 예상된다.

첫째, 뉴로심볼릭(Neuro-Symbolic) 접근의 확대가 주목된다. 기호적 계획이 제공하는 논리적 추론 능력과 신경망 기반 학습의 일반화 능력을 결합하는 방식은 복잡한 제약 조건과 불확실성을 동시에 처리할 수 있는 잠재력을 지닌다. 이는 산업 현장에서 요구되는 신뢰성과 유연성을 동시에 달성하는 데 중요한 역할을 할 것으로 기대된다.

둘째, 그래프 신경망(GNN)과 강화학습(DRL) 기반 통합 기법은 앞으로도 활발히 연구될 것이다. 실제 공정 계획과 스케줄링은 본질적으로 네트워크 구조를 띠기 때문에, 이를 그래프 형태로 모델링하고 학습하는 접근은 산업 적용 가능성이 크다. 이미 GNN과 DRL을 결합한 연구들이 통합 최적화 문제에서 효과를 보였으며, 향후에는 대규모 산업 데이터셋과 결합해 더욱 정교한 의사결정 지원이 가능할 것이다.

셋째, 대규모 언어모델(LLM)의 활용이 새로운 패러다임으로 떠오르고 있다. LLM은 자연어로 표현된 목표와 제약을 자동으로 형식화된 계획 및 스케줄링 모델로 변환할 수 있어, 전문가가 아니더라도 복잡한 문제 정의가 가능하다. 또한, LLM을 기존 최적화·학습 기반 기법과 결합하면, 실행 계획의 신뢰성과 안정성을 동시에 확보할 수 있다. 다만 LLM의 환각(Hallucination), 일관성 부족 문제는 반드시



해결해야 할 과제다.

넷째, 실시간 대응과 적응적 계획 · 스케줄링 연구가 더 강조될 것이다. 스마트 팩토리, 자율주행, 클라우드 서비스와 같은 응용에서는 환경 변화가 빈번히 발생하므로, 계획-스케줄링 시스템이 이를 빠르게 감지하고 동적으로 재계획(Replanning) · 재스케줄링(Rescheduling)을 수행해야 한다. 이를 위해 온라인 학습, 강화학습 기반 정책 전환, 시뮬레이션 기반 예측 기법이 결합될 필요가 있다.

마지막으로, 설명 가능성과 신뢰성 확보가 산업 현장 적용의 핵심 조건이 될 것이다. 딥러닝 기반 접근은 블랙박스적 특성으로 인해 결과 해석이 어렵지만, 실제 산업에서는 의사결정의 근거를 명확히 설명해야 한다. 따라서 eXplainable AI(XAI), 시각화 도구, 사용자 친화적 인터페이스와 같은 보완 기술이 병행되어야 한다.

## VIII. 결론

Task Planning과 Job Scheduling은 지능형 시스템의 핵심 기술로, 각각 “무엇을 할 것인가”와 “언제 · 어떻게 실행할 것인가”를 다루며 오랫동안 독립적으로 발전해 왔다. 본고에서는 두 기술의 정의와 관계를 정리하고, 전통적 기법에서 학습 기반 접근을 거쳐 최근의 대규모 언어모델(LLM) 기반 연구에 이르기까지 기술 발전 동향을 살펴보았다. 또한, 두 기술을 통합적으로 해결하려는 연구(IPPS)의 진전과

제조 · 클라우드 · 로봇틱스 등 다양한 산업 분야에서의 적용 현황을 검토하였다.

분석 결과, Task Planning은 기호 기반 탐색과 제약 충족에서 출발하여 강화학습, 뉴로심볼릭, LLM 기반 접근으로 확장되고 있으며, Job Scheduling은 전통적 최적화 · 휴리스틱 기법을 넘어 메타휴리스틱, GNN, DRL, 그리고 LLM 접목 가능성까지 진화하고 있음을 확인하였다. 두 기술은 개념적으로 차이를 가지지만, 실제 산업 응용에서는 상호 보완적 역할을 수행하며, 통합적 접근을 통해 현실적이고 효율적인 의사결정이 가능함을 보여주었다.

그러나 불확실성과 동적 환경 대응, 복잡한 제약 조건의 처리, 학습 기반 접근의 높은 데이터 비용, 설명 가능성과 신뢰성 부족 등은 여전히 산업 적용의 주요 과제로 남아 있다. 이러한 한계는 향후 기술 발전 방향에서 제시한 바와 같이, 뉴로심볼릭 융합, GNN · DRL 기반 통합 최적화, LLM을 통한 자동 모델링, 실시간 적응형 시스템, 그리고 설명 가능한 AI 기술을 통해 점차 해소될 것으로 기대된다.

### 용어해설

**PDDL** Planning Domain Definition Language의 약자로, 자동 계획(Automated Planning) 문제를 기술하기 위한 표준 언어. 1998년 International Planning Competition(IPC)를 계기로 제안되었으며, 현재까지도 AI Planning 연구의 사실상 표준 표현 방식으로 활용됨

**IPPS** Integrated Process Planning and Scheduling의 약자로, 제품 생산을 위한 공정 계획(Process Planning)과 작업 스케줄링(Job Scheduling)을 통합적으로 최적화하는 문제를 의미함

## 참고문헌

- [1] D.Z. Chen et al., "AI Planning: A Primer and Survey (Preliminary Report)," arXiv preprint, 2024. doi: 10.48550/arXiv.2412.05528
- [2] T. Anthony et al., "Thinking Fast and Slow with Deep Learning and Tree Search," in Proc. Int. Conf. Neural Inf. Process. Syst. (Long Beach, CA, USA), Dec. 2017, pp. 5366-5376.
- [3] P. Cao et al., "Large Language Models for Planning: A Comprehensive and Systematic Survey," arXiv preprint, 2025. doi: 10.48550/arXiv.2505.19683
- [4] M. Tantakoun et al., "LLMs as Planning Modelers: A Survey for Leveraging LLMs to Construct Automated Planning Models," arXiv preprint, 2025. doi: 10.48550/arXiv.2503.18971
- [5] V. Pallagani et al., "On the Prospects of Incorporating Large Language Models (LLMs) in Automated Planning and Scheduling (APS)," in Proc. Conf. Autom. Plann. Sched., (Alberta, Canada), Jun. 2024, pp. 432-444.
- [6] D.E. Smith et al., "Bridging the Gap Between Planning and Scheduling," Knowl. Eng. Rev., vol. 15, no. 1, 2000, pp. 47-83.
- [7] H. Xiong et al., "A survey of job shop scheduling problem: The types and models," Comput. Oper. Res., vol. 142, no. C, 2022.
- [8] A. Agnetis et al., "Fifty years of research in scheduling — Theory and applications," Eur. J. Oper. Res., vol. 327, no. 2, 2025, pp. 367-393.
- [9] K. Guo et al., "Integrated optimization of process planning and scheduling problems based on complex networks," J. Ind. Inf. Integr., vol. 36, 2023.
- [10] A. Jayanetti et al., "Reinforcement Learning based Workflow Scheduling in Cloud and Edge Computing," arXiv preprint, 2024. doi: 10.48550/arXiv.2408.02938
- [11] Y. Mei et al., "Learn to optimise for job shop scheduling: A survey with comparison between genetic programming and reinforcement learning," Artif. Intell. Rev., vol. 58, no. 160, 2025.
- [12] I.G. Smit et al., "Graph neural networks for job shop scheduling problems: A survey," Comput. Oper. Res., vol. 176, 2025.
- [13] H. Li et al., "Solving Integrated Process Planning and Scheduling Problem via Graph Neural Network Based Deep Reinforcement Learning," arXiv preprint, 2024. doi: 10.48550/arXiv.2409.00968