

**课程实验报告**

|  |  |
| --- | --- |
| 课程名称 | 自动规划 |
| 实验名称 | 电梯调度规划实验 |

|  |  |
| --- | --- |
| 学 号： | 58122231 |
| 姓 名： | 陆文韬 |
| 学 院： | 人工智能学院 |
| 专 业： | 人工智能专业 |
| 指导教师： | 张志政 |
| 实验日期： | 2024-2025秋季学期 |

1. 规划问题定义
2. 流的描述：

本实验构建的电梯调度系统是一个典型的多智能体运输规划问题。在这个系统中，我们需要协调多部电梯来高效地运送乘客到达他们的目标楼层。这个问题的核心在于优化资源分配和路径规划，同时最小化总运输成本。

从流的角度来看，系统中存在两种主要的流动实体。首先是电梯的移动流，电梯在不同楼层之间垂直移动，每次移动消耗基础成本。其次是乘客的运输流，当乘客搭乘电梯时，他们的位置会随着电梯一起改变，这种载人移动会产生额外的运输成本。

整个系统的状态空间由电梯位置、乘客位置、乘客搭乘状态以及目标达成状态等多个维度构成。状态转换通过四种基本动作实现，形成了一个有向状态图，其中每条边代表一个可能的动作执行，边的权重对应该动作的执行成本。

1. 动作描述：

系统设计了四种基本动作来实现完整的电梯调度功能。

**登梯动作（board）**：这个动作允许乘客进入电梯。执行的前提条件是乘客和电梯必须在同一楼层，乘客当前没有在任何电梯内，且乘客还未达成其运输目标。动作执行后，乘客的状态会更新为在指定电梯内。

**下梯动作（exit）**：这个动作让乘客离开电梯。前提条件包括乘客必须在电梯内，且乘客当前位置必须等于其目标楼层。执行后，乘客离开电梯并被标记为已达成目标。

**上升动作（ascend）**：电梯向上移动一层的动作。前提条件较为复杂，需要满足以下任一条件：存在未搭乘的乘客在更高楼层、存在已搭乘但目标楼层更高的乘客、或者电梯需要前往更高的目标楼层。同时电梯不能超过最高楼层限制。

**下降动作（descend）**：电梯向下移动一层的动作。前提条件与上升动作对称，需要存在位于更低楼层的服务需求，且电梯不能低于最低楼层。

每个动作的执行都会产生相应的成本。电梯移动产生固定的单位成本，而载人移动则会产生额外的倍数成本，这种设计反映了实际场景中载人运输的额外能耗。

1. 初始环境：

实验设置了一个五层楼的建筑环境，配备两部电梯和三名乘客。

电梯的初始配置为：lift\_a位于第3层，其目标位置设定为第3层；lift\_b位于第2层，目标位置也设定为第2层。这种设置意味着电梯在没有运输任务时会倾向于停留在当前位置。

乘客的初始分布具有一定的策略性：alex位于第1层，希望前往第4层，这是一个向上的长距离运输需求；beth位于第3层，希望前往第2层，这是一个向下的短距离需求；charlie位于第5层，希望前往第2层，这是一个向下的中等距离需求。

成本计数器初始化为零，最高楼层限制设定为5层。这种初始配置创造了一个需要精心规划的运输场景，其中不同的乘客请求在空间分布和方向上都有所差异。

1. 目标环境：

目标环境的定义采用了以结果为导向的方式，专注于运输任务的完成而非具体的位置约束。

具体而言，目标状态要求所有三名乘客都必须达成其个人的运输目标。也就是说，alex必须成功到达第4层，beth必须到达第2层，charlie也必须到达第2层。当乘客到达目标楼层并执行下梯动作后，系统会自动标记该乘客为"goal\_reached"状态。

这种目标定义的优势在于它不会过度约束电梯的最终位置，给规划算法更大的灵活性来找到成本最优的解决方案。电梯在完成所有运输任务后可以停留在任何合理的位置，只要总成本最小。

1. 规划生成机制（规划算法）介绍：

本实验使用A\*搜索算法结合LM-Cut启发式函数来求解规划问题。这是一种在自动规划领域广泛应用的最优搜索方法。

A\*算法是一种图搜索算法，它维护一个评估函数f(n) = g(n) + h(n)，其中g(n)表示从初始状态到当前状态n的实际成本，h(n)是从状态n到目标状态的启发式估计成本。算法优先展开具有最小f值的节点，这样能够保证在启发式函数可采纳的前提下找到最优解。

LM-Cut（Landmark Cut）启发式是一种基于地标的启发式函数。地标是指在任何有效计划中都必须为真的命题。LM-Cut通过分析动作之间的依赖关系来识别这些地标，然后计算达成这些地标所需的最小成本作为启发式值。这种启发式函数具有可采纳性和一致性，能够有效指导搜索过程。

在数值规划的上下文中，LM-Cut需要处理数值变量和数值效果。它通过分析数值前提条件和效果之间的因果关系来构建地标，特别适合处理本实验中涉及的位置变量和成本优化问题。

1. 规划程序设计与实现
2. 介绍

本实验的规划程序使用PDDL（Planning Domain Definition Language）语言编写，这是自动规划领域的标准建模语言。程序分为领域文件（domain）和问题文件（problem）两部分。

领域文件定义了问题的通用结构，包括类型系统、谓词、函数和动作模式。类型系统将对象分为电梯（lift）和乘客（person）两类，这种层次化的类型定义有助于约束动作参数和提高搜索效率。谓词部分定义了系统的布尔状态变量，如乘客是否在电梯内、是否已达成目标等。函数部分定义了数值状态变量，包括位置信息、目标信息和成本计数器。

动作定义是程序的核心部分。每个动作都包含参数声明、前提条件和效果描述。前提条件使用一阶逻辑表达式来描述动作的可执行条件，效果部分则描述动作执行后的状态变化。程序广泛使用了条件效果（conditional effects）和全称量词（universal quantification）来处理复杂的状态更新逻辑。

问题文件则定义了具体的问题实例，包括对象声明、初始状态和目标状态。初始状态通过等式来设定各种数值变量的初值，目标状态使用逻辑表达式来描述期望达成的条件。

1. 运行情况说明
   1. 输入输出说明：

程序的输入包括两个PDDL文件：领域文件（lift-transport.pddl）和问题文件（transport-scenario.pddl）。领域文件描述了电梯调度问题的通用模型，而问题文件定义了具体的实例参数。

运行命令的格式为：./fast-downward.py domain.pddl problem.pddl --search "astar(lmcut())"，其中指定了使用A\*搜索算法和LM-Cut启发式函数。

程序的输出包括多个阶段的信息。首先是解析阶段，系统会分析PDDL文件的语法和语义正确性。然后是预处理阶段，包括变量实例化、动作展开和可达性分析。最后是搜索阶段，输出搜索统计信息和找到的计划序列。

成功执行时，程序会输出一个动作序列，每个动作包含动作名称和具体参数。同时会报告计划的总成本、搜索的节点数量、评估的状态数量等统计信息。

* 1. 主要数据结构说明：

在PDDL层面，程序使用了多种数据结构来表示系统状态和约束关系。

谓词（predicates）用于表示布尔状态变量。例如，(inside ?person ?lift)表示乘客是否在电梯内的二元关系，(goal\_reached ?person)表示乘客是否已达成目标的一元谓词。

函数（functions）用于表示数值状态变量。(person\_location ?person)和(lift\_position ?lift)分别表示乘客和电梯的当前位置，(destination ?person)表示乘客的目标位置，(total\_cost)表示累积的总成本。

在规划器的内部实现中，状态被表示为变量赋值的集合。每个状态包含所有谓词变量的真值和所有函数变量的数值。动作被表示为状态转换函数，包含前提条件的检查逻辑和效果应用的更新逻辑。

搜索空间被组织为一个隐式的状态图，其中节点表示状态，边表示动作。搜索算法维护开放列表和封闭列表来管理待探索和已探索的状态。

* 1. 运行案例：

考虑一个具体的执行案例。在初始状态下，lift\_a在3层，beth也在3层且希望前往2层。首先执行动作(board beth lift\_a)，使beth进入电梯。接着执行(descend lift\_a)，电梯和beth一起移动到2层。由于beth到达了目标位置，可以执行(exit beth lift\_a)让beth下电梯，此时beth被标记为goal\_reached状态。

这个简单的案例展示了动作序列如何协调地改变系统状态。每个动作的执行都需要检查复杂的前提条件，并可能触发多个条件效果。例如，descend动作不仅改变了电梯位置，还更新了所有搭乘乘客的位置和累积成本。

完整的求解过程涉及多个这样的子任务，规划器需要找到一个动作序列来依次完成所有乘客的运输需求，同时最小化总成本。

1. 分析
2. 通用性说明（从算法和规划语言两方面说明）：

从算法角度来看，A搜索结合LM-Cut启发式具有很强的通用性。这种组合不仅适用于本实验的电梯调度问题，还可以应用于广泛的数值规划问题，如物流运输、资源调度、机器人导航等领域。A算法的通用性体现在它对问题域无关的搜索策略，而LM-Cut启发式的通用性则体现在它能够自动从PDDL描述中提取有用的启发信息。

从规划语言角度来看，PDDL提供了强大的表达能力。本实验使用的PDDL特性包括类型系统、数值函数、条件效果、存在量词和全称量词等，这些特性使得PDDL能够建模复杂的现实世界问题。特别是数值规划的支持使得系统能够处理连续变量和优化目标，这对于许多实际应用是必不可少的。

程序的模块化设计也增强了通用性。通过分离领域定义和问题实例，同一个领域模型可以用于不同规模和配置的电梯调度问题。用户只需修改问题文件中的对象数量、初始状态和目标状态，就可以生成不同的问题实例。

1. 完备性说明：

A\*搜索算法在理论上是完备的，即如果存在解，算法一定能够找到解。这种完备性的保证基于以下几个条件：搜索空间是有限的或者满足适当的扩展条件、启发式函数是可采纳的、状态表示是完整和一致的。

在本实验的电梯调度问题中，状态空间虽然很大但是有限的。电梯和乘客的位置都被限制在有限的楼层范围内，乘客的搭乘状态和目标达成状态也是有限的。因此，理论上存在一个有限的状态图包含所有可能的系统状态。

LM-Cut启发式函数被证明是可采纳的，即它永远不会高估从当前状态到目标状态的实际成本。这个性质确保了A\*搜索能够找到最优解。

然而，实际的完备性还受到计算资源限制。如果问题规模过大，搜索可能会因为内存不足或时间超限而终止。在这种情况下，算法的理论完备性无法在实践中实现。

1. 计算复杂度说明：

电梯调度问题的计算复杂度分析需要考虑多个维度。

状态空间的大小是复杂度分析的关键。假设有E部电梯，P名乘客，F层楼，那么单纯的位置组合就有F^(E+P)种可能。加上乘客的搭乘状态（每个乘客可以在任一电梯内或不在电梯内），状态数量达到F^(E+P) × (E+1)^P。再考虑目标达成状态，总的状态空间大小约为F^(E+P) × (E+1)^P × 2^P。

在最坏情况下，A\*算法可能需要探索整个状态空间，因此时间复杂度为指数级。然而，好的启发式函数能够显著减少实际探索的状态数量。LM-Cut启发式通过提供准确的成本估计来指导搜索，大大提高了搜索效率。

空间复杂度主要由开放列表和封闭列表的大小决定，在最坏情况下也是指数级的。但实际运行中，有效的状态表示和内存管理技术可以显著降低内存需求。

值得注意的是，这种复杂度分析是基于最坏情况的理论分析。在实际的电梯调度问题中，由于物理约束和实际需求的特点，搜索空间的结构通常比理论分析更加友好，使得实际求解时间远小于理论上界。

1. 正确性说明：

程序的正确性可以从多个层面进行验证。

首先是语义正确性。PDDL模型必须准确反映现实世界的电梯调度逻辑。这包括动作的前提条件是否合理、效果是否完整、约束是否充分等。例如，board动作的前提条件确保乘客只能在与电梯相同的楼层登梯，且不能同时在多部电梯内。这些约束反映了物理世界的基本规律。

其次是算法正确性。A\*搜索算法的正确性有严格的理论保证，在启发式函数可采纳的前提下，算法保证找到最优解。LM-Cut启发式的可采纳性也有理论证明，确保了算法的正确性。

还有实现正确性。PDDL代码必须正确实现设计的逻辑。这包括语法正确性、类型一致性、量词使用的正确性等。通过PDDL解析器的语法检查和语义分析，可以发现大部分实现错误。

最后是结果正确性。产生的计划必须是可执行的和有效的。这要求每个动作在执行时都满足其前提条件，执行后能够正确更新状态，最终状态满足目标条件。可以通过计划验证工具来自动检查这种正确性。

1. 最优性：

本实验使用的求解方法能够保证找到最优解，这种最优性保证来自A\*算法和LM-Cut启发式的理论性质。

A算法的最优性基于其评估函数f(n) = g(n) + h(n)的性质。当启发式函数h(n)是可采纳的（即不高估实际成本）时，A算法保证首次到达目标状态时找到的解是最优的。这是因为算法总是优先展开具有最小f值的节点，而可采纳的启发式确保了真正的最优路径不会被错误地剪除。

LM-Cut启发式函数具有可采纳性，这意味着它提供的成本估计永远不会超过实际的最小成本。更进一步，LM-Cut还具有一致性（单调性），即启发式值满足三角不等式，这进一步增强了搜索的效率和正确性。

在电梯调度问题的上下文中，最优性指的是找到总成本最小的运输方案。成本函数包括电梯移动的基础成本和载人移动的额外成本，这种设计反映了实际场景中的能耗考虑。

需要注意的是，这种最优性是相对于给定的成本函数而言的。如果成本函数的设计不能准确反映实际的优化目标，那么数学上的最优解可能不是实际意义上的最优解。因此，成本函数的合理设计对于获得有意义的最优解至关重要。

1. 不足说明：

尽管本实验的设计和实现具有多方面的优势，但仍存在一些不足之处需要改进。

首先是模型简化的问题。为了保持问题的可解性和复杂度的可控性，模型做了许多简化假设。例如，电梯移动被假设为瞬时完成，没有考虑加速、减速和停靠时间；乘客的登梯和下梯也被假设为瞬时动作，忽略了实际的时间消耗。这些简化使得模型与现实情况存在一定差距。

其次是扩展性的限制。当电梯数量、乘客数量或楼层数量显著增加时，状态空间的爆炸性增长会使得求解变得非常困难。虽然启发式函数能够在一定程度上缓解这个问题，但对于大规模的实际应用，可能需要采用近似算法或分层规划方法。

另外，目标函数的单一性也是一个限制。当前的模型只考虑了成本最小化，没有考虑其他可能的优化目标，如乘客等待时间、服务公平性、电梯负载均衡等。在实际应用中，通常需要在多个目标之间进行权衡。

最后，模型的动态性不足。当前的规划是基于静态的初始信息进行的，没有考虑动态到达的新乘客请求或突发的设备故障。实际的电梯调度系统需要能够处理动态变化的环境和实时的重规划需求。

这些不足为未来的研究和改进提供了方向，包括更精确的时间建模、多目标优化、分层规划方法、以及动态重规划技术等。