

# 静若处子，动若脱兔——时窗对抗采集站的移动资源分配问题的有效表征分析与研究

叶臻铭, 12011404

南方科技大学

12011404@mail.sustech.edu.cn

## Abstract

国际空间优化大赛（SpOC）是欧洲前沿概念研究组（ACT）举办的高水平优化赛事，每年都吸引了来自全球各地的参与者。其中，我们认为2022赛季中的第三题Resource Delivery具有较高的难度、一定的现实意义和良好的应用前景，值得深入研究。本文首次系统研究了该优化问题，并将该问题命名为**时窗对抗采集站的移动资源分配（Mobile Resource Delivery for Time Competing Stations, MRDTCS）**问题。

对于优化问题，合适的表征是对于问题的有效求解是至关重要的，因此本文为MRDTCS提出了几种形式化方法组合，包括**两种基本视角、两种决策思路和两种求解目标**。在这几种等价的问题形式化中，可以证明MRDTCS是NP难问题，精确求解方法只能用于较小的问题实例。因此，本文针对不同的表征方式提出了几种基本的启发式算法和元启发式算法，并且通过实验分析了这些形式化方法的有效性。初步研究表明，**采集站视角下的单目标静态形式化**具有较好的表征效果。

针对这一表征方式，本文提出一个新颖的元启发式算法，称为**静若处子-动若脱兔（Simultaneous Static and Dynamic Programming, SSDP）**，旨在结合**演化计算（Evolutionary Computing, EC）**和**强化学习（Reinforcement Learning）**两大前沿人工智能技术来优化MRDTCS问题的不同部分。实验表明该算法可以快速有效地找到高质量的解，可以作为后续MRDTCS问题研究的基线算法，对于其他最优化问题的求解也有一定的意义。

**关键词：**演化计算、动态规划、强化学习、多目标优化

\* 本文的Python版本代码已在[GitHub](#)上开源。

## Contents

静若处子，动若脱兔——时窗对抗采集站的移动资源分配问题的有效表征分析与研究

Abstract

Contents

1. Introduction

## 2. Related Work

### 2.1 演化计算

### 2.2 动态规划与强化学习

## 3. Effective formalization of acquisition allocation optimization problem for space station

### 3.1 什么样的表征方式是好的?

#### 3.1.1 什么是表征方式? 为什么解的表征很重要?

#### 3.1.2 难以违反约束的解表征方式是好的

### 3.2 表征约简: 空间站视角与小行星视角

#### 3.2.1 空间站视角: 水滴切换问题

#### 3.2.2 小行星视角: 四维碎片守护问题

### 3.3 表征选择: 动态规划与静态规划

### 3.4 表征选择: 单目标优化与多目标优化

## 4. Methodology

### 4.1 初值设置: 基于探索性数据分析的几种启发式算法

#### 4.1.1 寿司转盘模型

#### 4.1.2 最优点目问题

材料均匀假设

### 4.2 演化计算

### 4.3 强化学习

## 5. Experiments

## 5. Conclusion

## 参考文献

---

# 1. Introduction

国际空间探测轨迹优化大赛 (GTOC) 是欧洲空间局 (ESA) 于2005年发起的高水平、专业性、国际性赛事, 是航天任务设计领域国际最高水平角逐的擂台, 被称为航天界的“奥林匹克竞赛”。2021年, 清华大学航天航空学院代表队在比赛中夺冠。<sup>1</sup> 2021年的该比赛针对美国物理学家提出的“戴森球”设想, 具体指出如何确定戴森环位置、如何设计小行星轨道等问题, 请求参赛团队设计一个可行、高效的技术方案。<sup>2</sup> 2022年, ESA下属的欧洲前沿概念研究组 (ACT) 与期刊 GECCO合作, 发起了国际空间优化大赛 (Space optimisation Competition, SpOC)。<sup>3</sup> 其中该比赛的第三题 Resource Delivery 虽然问题描述的场景不同, 但是可以看做是2021年GTOC比赛的戴森球问题的后继问题。

戴森环是一种包裹太阳的结构, 用于充分利用太阳的辐射能量, 从而加速人类文明的等级跨越。如图所示, 戴森环由12个建造中的戴森球 (又称Building Station) 组成。戴森球的持续建造需要大量的材料, 2021 GTOC 比赛中, 清华大学等团队从83453颗候选小行星中选择了大约三百多颗小行星作为戴森球的材料来源, 设计了小行星运动的轨道。<sup>3</sup> 在已经确认了小行星子集、轨迹和戴森

球位置的前提下，SPoC中的Resource Delivery 问题试图解决如何将材料均匀分配给各个戴森球的问题。因为戴森环的建成需要所有的戴森球的建成，不能只有一个戴森环获得大量的材料。

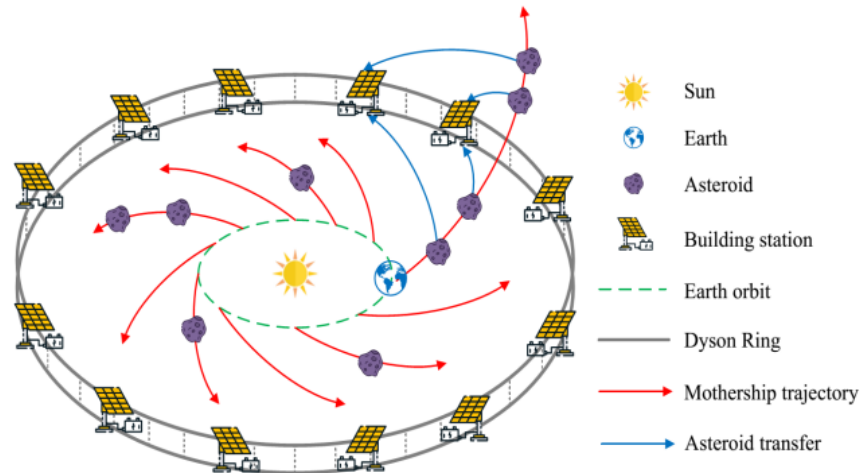


Figure 1 Illustration of the construction of the “Dyson ring”.

本文将SPoC中的Resource Delivery问题规范命名为时窗对抗采集站的移动资源分配 (**Mobile Resource Delivery for Time Competing Stations, MRDTCS**) 问题，相关概念解释如下。

- 采集站 (Station, 又称空间站)。即GTOC问题中的12个戴森球，是采集资源的单位。
- 资源点 (Resource, 又称小行星)。即GTOC问题中戴森球的资源来源——小行星。
- 时窗对抗 (Time Competing) 。
  - 首先，不同采集站之间呈现一种对抗关系。
    - 小行星是有限的，一旦被一个空间站收集，就不能被另外一个空间站
    - 优化问题的目标是使得所有采集站中采集量最少的采集站的采集量最大。
  - 其次，由于能量的限制，同一时间段只有一个空间站是激活的，而且每个空间站只能激活一段连续时间窗口。
    - 这与传统的Resource Delivery问题不同，增加了时窗的限制。
- 移动资源分配 (Mobile Resource Delivery)
  - 小行星是材料的来源，也是被收集的资源。
  - 与传统的Resource Delivery问题不同，小行星是不断移动的(Mobile)，当超出一定的距离之后，就无法被采集站捕获。
  - 小行星的运动是周期性的轨迹运行，因此如果不被捕获的话，小行星被采集站错过后会反复经过其他空间站的捕获范围，然后再次与一开始的采集站相遇。

MRDTCS的具体定义如下。

- 问题实例表征 (Problem Instance Representation)
  - 问题的实例可以定义为资源采集机会的集合，可以存储在关系型数据库中。

- 问题实例I定义为  $I = \{(t_i, s_i, r_i, v_i)\}_{i=1}^{I^N}$
- 其中
  - $t_i \in [0, 80] \cap \mathbb{R}$  是采集机会发生的时间点（资源的采集时间忽略不计）
  - $s_i \in [1, 12] \cap \mathbb{Z}$  是采集站编号， $r_i \in [1, 340] \cap \mathbb{Z}$  是资源点的编号。
  - $v_i \in \mathbb{R}^+$  是资源的具体数值，数值越大越好。
  - I中的一条record代表了带有  $v_i$  的资源  $r_i$  在  $t_i$  时间点可以被采集站  $s_i$  采集。
- 问题解表征（Problem Solution Representation）
  - 问题的解  $x$  分为两部分，采集站时间窗口(Time Windows)和资源点实际采集机会分配(Opportunity Delivery),  $x = (T, D)$
  - 时间窗口T定义为  $T = [T_{1i}, T_{1f}, \dots, T_{12i}, T_{12f}]$ , 其中  $T_{ji}$  和  $T_{jf}$  分别表示编号为j的采集站的开始时间和结束时间。
  - 资源点实际采集机会分配D定义为
- 问题优化目标（Optimization Objective）
  -

SpOC比赛中，还增加了一些额外的约束。

- 一个资源点的资源数值  $v_i = (a_i, b_i, c_i) \in \mathbb{R}^3$ 
  - 对于任意两个不同采集站收集资源总和的  $V_1$  与  $V_2$ ，定义  $\mathbb{R}^3$  上的非偏序关系P
  - $V_1 P V_2 \iff \min(V_1) \leq \min(V_2)$
  - 该关系自反、传递，但是不满足反对称。
  - SpOC要求在P的意义上目标达到最大。
  - 这类型的问题是MRDTCS的一个特例，我们称为竞争多资源分配问题。

本文是首次提出MRDTCS问题并且进行分析，此前针对该问题没有相关的算法。

## 2. Related Work

### 2.1 演化计算

演化计算是一种基于生物进化原理的优化方法，它通过模拟自然界中的进化过程，不断地从种群中筛选出适应度高的个体，并通过遗传操作（交叉、变异等）产生新的个体，最终得到最优解。演化计算在最优化理论与方法中占有重要地位，它可以用于求解各种复杂的优化问题，如函数优化、组合优化、约束优化等。演化计算与其他最优化方法相比，具有全局搜索能力强、鲁棒性好、易于并行处理等优点。 4

## 2.2 动态规划与强化学习

动态规划的核心思想是将一个大问题分解成若干个小问题，通过寻找子问题的最优解可以得到问题的最优解，并且可以找到子问题状态之间的递推关系。与贪心算法相对，动态规划不只是通过一步观察的启发式信息，而是结合了子问题的信息；与分治相对，动态规划解决的是重叠子问题。<sup>5</sup> 只要定义状态空间，动态规划可以解决大部分的多阶段决策问题，然而当状态空间的定义达到指数级别时，动态规划便难以在有限时间内给出有效的解。

强化学习是一种机器学习方法，它通过智能体与环境的交互来学习如何做出决策。强化学习主要是为了解决动态规划中的两个问题：

- 通过神经网络来解决拟合值函数、Q函数和策略函数的问题，因为空间可能是连续的，或者如上文所说状态空间太大了，无法直接动态规划求解。
- 通过采样来解决状态转移具有随机性的问题。比如智能机器人中状态转移往往不是确定的。

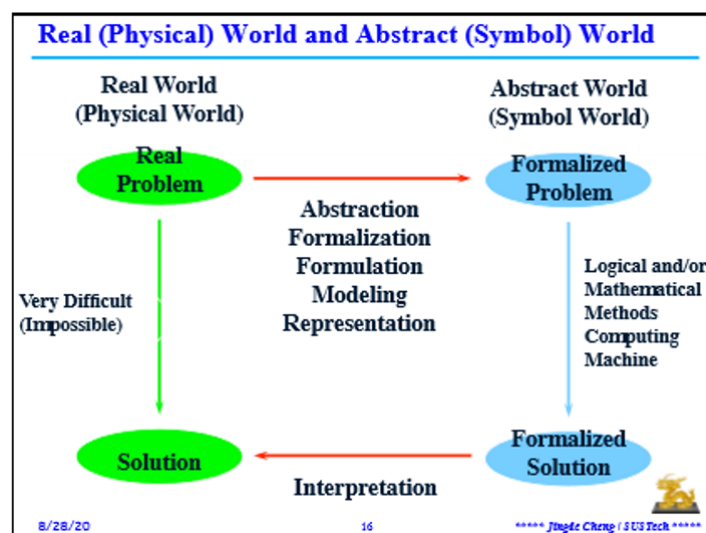
6

## 3. Effective formalization of acquisition allocation optimization problem for space station

### 3.1 什么样的表征方式是好的？

#### 3.1.1 什么是表征方式？为什么解的表征很重要？

解表征方式（Representation of Solution）是待求解问题的解的表达形式。如图所示，计算机科学是建立可计算模型<sup>7</sup>来求解现实问题的艺术，使用计算机科学求解问题的过程是将原问题表达为形式化问题输入、求解形式化问题、将形式化问题的输出翻译为原问题的解的过程。解表征方式的确定就是决定问题形式化的过程中如何表达问题的输出。计算机科学也是多层抽象的过程，问题被不断翻译为更加好求解的问题，再不断地返回结果给上一层抽象。



在演化计算研究领域，**待求解问题**与**问题的解决方法**是两个不同研究重点，从大的方面来分，演化计算可以解决优化问题，以及通过优化问题来解决建模问题（即机器学习或者模式识别问题）和仿真问题。<sup>8</sup>而就优化问题而言，可以根据**目标函数、问题约束的存在与否**将待求解问题分为**约束优化问题（COP）、约束满足问题（CSP）、自由优化问题（FOP）**和非优化问题。相同的现实问题往往通过不同的**问题形式化**（其中包括对解的表征方式的形式化）成为不同类型的问题，因此也有不同的**求解难度**，比如<sup>8</sup>

- 八皇后问题
  - 可以形式化为CSP：棋盘上摆放8个皇后，使得满足任意两个皇后不能互相攻击
  - 可以形式化为FOP：优化使得皇后之间互相攻击的数量最少

Eiben和Smith指出，要想成功运用演化计算来解决问题，较好的实践应当是针对特定的问题类型进行设计的，提出“技巧GA”必定是在一定的问题表示下才能奏效。<sup>8</sup>八皇后问题实际上就是一个很简单但是能清晰说明这一点的例子。

- 八皇后问题的几种解表征方式
  1. 表示为8个皇后棋盘上坐标的集合。
  2. 注意到皇后攻击的特点，每一行必定有一个皇后，于是表示为每一行皇后所在列数的列表。
  3. 注意到皇后攻击的特点，每一列也只能有一个皇后，其实皇后所在列的是互斥的，所以可以表示为一个8维permutation。

这四种解的表征方式完全不同，他们的求解难度也截然不同。我们可以注意到，第三种表示方式比前面三种，有效的消除了约束条件，使得可行域的范围更加清晰可见。解的表示本身决定了解搜索空间的大小，第三种表示方式明显降低了搜索空间大小，减少了算法运行中大量的无效搜索，因此也更容易求解。

### 3.1.2 难以违反约束的解表征方式是好的

我们当然知道，可以简单的把约束转换成优化目标，违反约束了就给一个惩罚函数。但是在本题中，原始的解表征是很差的一种表示方式。仅仅知道违反了约束，很难通过有效的算子变为有约束问题。

一般处理约束有几种办法：

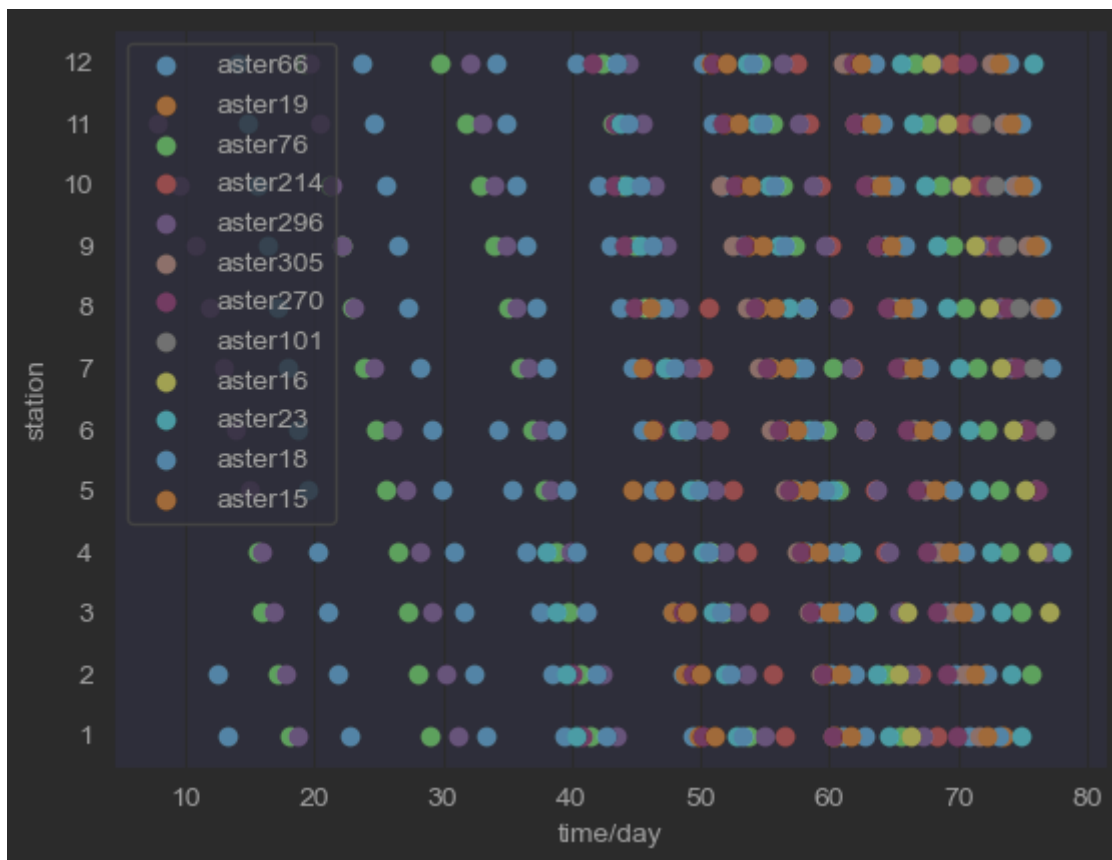
- 罚函数法
- 修复函数法
- 可行域限制搜索法

## 3.2 表征约简：空间站视角与小行星视角

在原问题中，解被表征为，我们很容易观察这实际上是冗余的：当小行星被谁采集

### 3.2.1 空间站视角：水滴切换问题

我们首先从空间站的视角来看待问题。对于一个空间站来说，随着时间的流逝，小行星们会一次次出现在空间站的面前。如图所示，横坐标是可行时间，纵坐标是Station编号。每一行上的点代表了该行对应空间站会遇到



面对这样一个视角，我们可以很容易的将问题简化为一个动态、单人的游戏，本文命名该游戏为“水滴（Water drop）切换问题”：

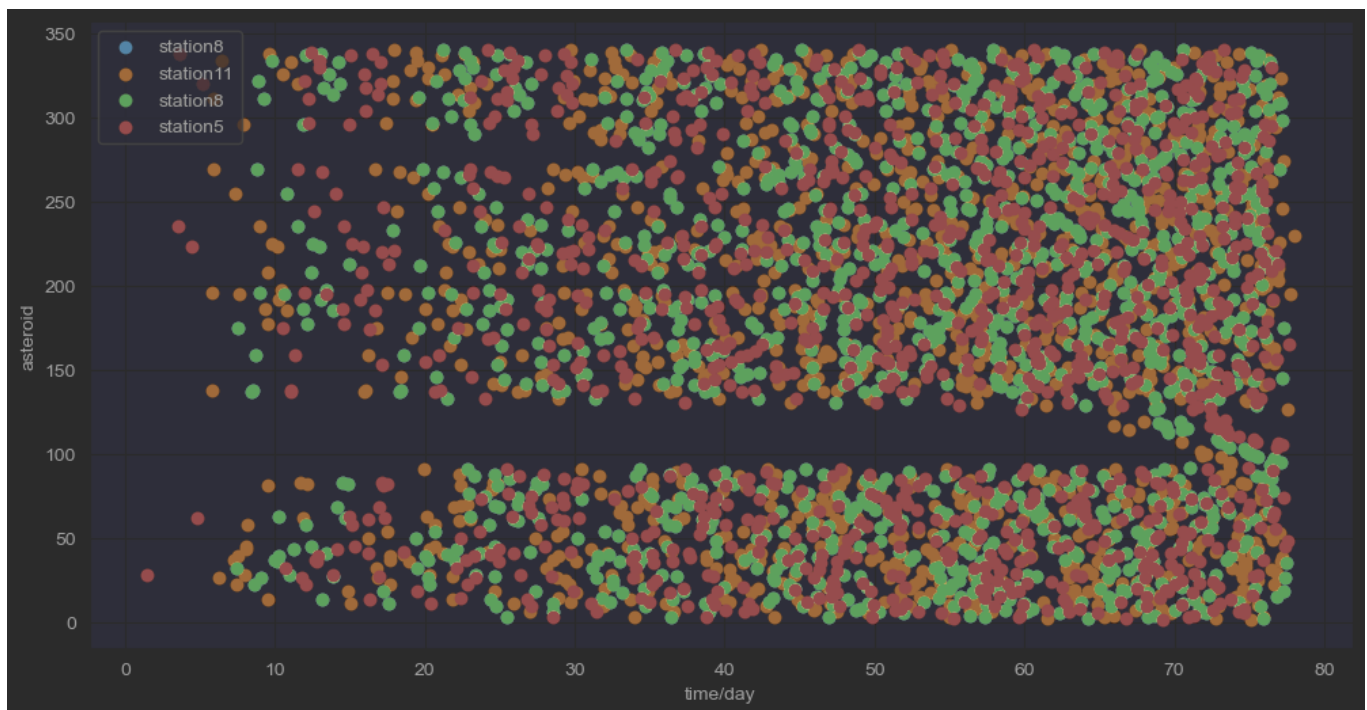
- 游戏背景
  - 由n条装备了先进技术的太空飞船组成的太阳系舰队（即原问题中的asteroids）排列在太空中，
  - 排列成宽12行长达80km的矩阵防御带，
  - 对迎接三体世界的“礼物”水滴的仪式严阵以待。
  - 然而，水滴却有一个不同寻常的使命——摧毁人类舰队。
- 游戏操控：你将扮演三体水滴执行官，实时控制一颗水滴上下移动（切换行）。
- 游戏目标：摧毁太阳系舰船可以获得ABC三种物质，需使在每一行采集最少的物质的行最少采集量最大。



- 游戏约束

- 当水滴上下切换行时，为了避免地球舰队还有有生力量，水滴的AI系统自动控制超强激光在离开这一行前摧毁这一行的所有舰船，激光远程摧毁的舰船的燃料无法被收集。
- 人类舰船之间存在量子通信，当一艘舰船被摧毁时，与之通信的舰船会得知水滴很厉害的信息，立刻开始逃逸，从而无法被水滴打击。
  - 智子已经提前探测了舰船之间的量子通信关系网络，发现有340个通信频道（即原问题的asteroids编号）。
  - 一条舰船只有一个量子通信频道，相同频道的舰船之间可以相互通信，不同频道的无法及时通信（只能通过电磁波通信，战争结束了才能收到信息）。
- 水滴在一行直线前进的距离不少于1km，才能切换行。

### 3.2.2 小行星视角：四维碎片守护问题



- 游戏背景

- 在一次探险中意外发现君士坦丁堡的真相，人类提前掌握了四维碎片的收集技术与使用方法，正在赶往太阳系路上的三体文明感到了严重威胁。
- 三体元首立刻下令派出**340颗水滴**进攻太阳系舰队，以便进一步摧毁位于木星上的四维碎片收集装置。然而，等待这些水滴的是地球舰队的四维碎片武器。
- 尽管人类掌握四维碎片的时间不久、相关技术还不成熟，针对**340个水滴**大敌来袭，四维碎片工厂还是紧急生产了**340块四维碎片**。游戏过程中四维碎片无法生产。
- 太阳系舰队排列成宽340行长达80km的矩阵防御带，**每一行舰队所有飞船中有且仅有一块四维碎片**，面对的敌人有且仅有一个水滴。



- 游戏操控：作为太阳系舰队四维碎片作战部司令，你需要**决策**每一行的那一块四维碎片资源分配给哪一条飞船。
- 游戏约束
  - 水滴经过装备了四维碎片武器的舰船时被消灭，材料被收集。
  - 水滴飞行过程中材料会消耗，**越早攻击水滴，你获得的材料越多。**
  - 四维碎片武器有**12种型号**，不同型号的使用方式不同，**每一条飞船只有一个型号的武器。**
  - 由于可能的四维坍缩问题，当相同型号的四维碎片武器放置四维碎片时，中间形成一条安全禁区带，禁区带内不能使用其他型号的四维武器，否则会产生共振。禁区带还会往两边**蔓延1km**。
- 游戏目标：消灭水滴可以收集ABC三种物质，需在每种型号的武器采集最少的物质的型号最少采集量最大。

### 3.3 表征选择：动态规划与静态规划

根据上面的两种基本视角，我们很容易将问题分为动态规划和静态规划两种形式。动态规划是多阶段最优决策中的一种重要方法，通过明确定义问题的**状态、行动和辅助信息**，列出状态转移方程（又称Bellman Equation），将最优决策问题的求解多个重叠子问题的求解。

以空间站视角下的水滴切换问题为例，

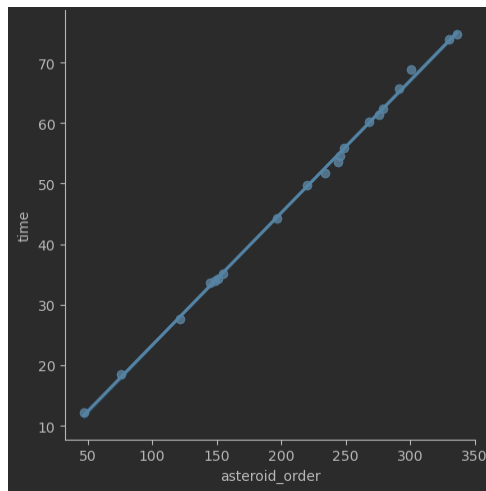
### 3.4 表征选择：单目标优化与多目标优化

MRDTCS的一个重要难点在于

## 4. Methodology

### 4.1 初值设置：基于探索性数据分析的几种启发式算法

#### 4.1.1 寿司转盘模型



#### 4.1.2 最优点目问题

材料均匀假设

#### 4.2 演化计算

#### 4.3 强化学习

端到端优化与交替优化

## 5. Experiments

## 5. Conclusion

# 参考文献

- 
1. 航天航空学院团队斩获2021年国际空间探测轨迹优化大赛冠军-清华大学. (n.d.). Retrieved April 14, 2023, from <https://www.tsinghua.edu.cn/info/1175/88685.htm> ↵
  2. 清华大学团队获航天界奥林匹克全球冠军，平均年龄 25 岁，如何评价这一成绩？ — 知乎. (n.d.). Retrieved April 14, 2023, from <https://www.zhihu.com/question/498075734>. ↵
  3. Team, A. C. (2022, March 1). GECCO 2022 Space Optimisation Competition (SPoC). <https://esa.int/gsp/ACT/projects/gecco-2022-competition/> ↵ ↵
  4. 最优化：建模、算法与理论/最优化计算方法 – pku.edu.cn. <http://faculty.bicmr.pku.edu.cn/~wenzw/optbook.html> 访问时间 2023/4/14. ↵
  5. Kleinberg, J., & Tardos, É. (2014). *Algorithm design* (Pearson new internat. ed. [der] 1.ed). Pearson. ↵
  6. Stuart J. Russell, Peter Norvig, 诺维格, 罗素, 祝恩和殷建平, 《人工智能:一种现代的方法》, 清华大学出版社, 2013, doi: 9787302331094. ↵
  7. 科学网—何谓“计算”? —可计算性理论简介—程京德的博文. (n.d.). Retrieved April 14, 2023, from <https://blog.sciencenet.cn/home.php?mod=space&uid=2371919&do=blog&id=1383722> ↵
  8. A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computation*. Springer-Verlag, Berlin, 2015. ↵ ↵ ↵