1. Demand：求解三维空间无碰撞的存续联通路径
2. Project Address：<https://github.com/bladesaber/MAPF_Pipeline>
3. Algorithm Structure:

算法组成：

1. 求解三维空间存续无碰撞路径
   1. 参考方案1: CBS[10] + A-star + CAT(Conflict Avoidance Table)[12]
   2. 参考方案2: Prioritized Planning[19, 20] + A-star

初步评价：方案1的求解路径为理论最短路径，但求解空间大，需要计算资源大，求解时间长。在方案初步结果确认后，可考虑添加Operator Decomposition[6,7,8]与Bypass[17]方法进行优化。方案2的求解路径为次优路径，但无法保证一定有求解结果。

1. 所得路径进行局部平滑
   1. 参考方案1：Hyprid A-star
   2. 参考方案2：Bezier Smoothing（以路径点为控制点）

初步评价：后续需要结合流体方面的函数设计

1. 管道效果可视化
   1. 参考方案1：基于PyVista库实现（参考https://docs.pyvista.org/）
   2. 参考方案2（候选）：基于Mayavi库实现
2. Plan：

2023-03-10：

1. 计划2023-04-01前完成算法组成（1）中两个方案（基于C++）。不确认在三维空间上A-star会不会由于搜索空间过大的问题导致崩溃，因此初期Grid空间设计为（50x50x50）先做尝试。
2. 之后需要一个粗糙的可视化代码（基于Python）。
3. 需要部分模拟应用场景的参数，对比以上两种算法求解该部分场景的Metric差异，以及对比人工设计与算法求解路径的差异（忽略局部路径平滑），已确认下一步是否继续进行。

2023-03-29:

我确认思路为以下两种

1. CBS+角度离散限制的Hybrid-Astar+梯度曲线平滑（可能不使用Dubins的启发式），一些很特别的问题需要注意：
   1. 障碍物的形状不再是cube，这就造成了基于Grid的启发式会失效
   2. 最小外接圆形成的CSC联通域一定大于其他外接圆形成的CSC联通域
2. CBS+Reactive方法（当障碍超出一定阈值使用A\*作为启发式）+Bezier曲线平滑，一些很特别的问题需要注意：
   1. CBS对单个节点的解决方案非常的敏感，如果单个节点的解决方案不是最优就会造成无效探索
   2. Grid启发式失效，必须想方法妥善解决U型局部最优点的问题
   3. 不适合于密集管道

（3）仿造ICTS，采用CBS+A-star+smoothing（当smoothing产生conflict则不接受当前grid得最短路径，继续搜索知道超出最优路径thresold后，搜索失败）

（4）我感觉要融合以上两种该方法，需要一个状态转换的方法。只要可以保证最优路径与Reactive的求解路径一定有重叠，就一定可以找到一个解。但仍然无法保证最优解。Paper Reference:

Survey：

1. A Survey of the Multi-Agent Pathfinding Problem, Erwin Lejeune
2. Multi-Agent Path Finding – An Overview , Roni Stern, 2022
3. Multi-Agent Pathfinding: Definitions, Variants, and Benchmarks, Roni Stern
4. Search-Based Optimal Solvers for the Multi-Agent Pathfinding Problem: Summary and Challenges(SoCS 2017)
5. T. Uras and S. Koenig. An Empirical Comparison of Any-Angle Path-Planning Algorithms. In Proceedings of the Annual Symposium on Combinatorial Search, 2015

MAPF：

Search Based Method:

1. WHCA-star：Silver, D. 2005. Cooperative pathfinding. In Artificial Intelligence and Interactive Digital Entertainment (AIIDE), 117–122
2. EPEA-star：Goldenberg, M., Felner, A., Sturtevant, N.R., Holte, R.C., Schaeffer, J.: Optimal generation variants of EPEA. In: SoCS (2013)
3. Independence Detection + OD：Standley, T. S. 2010. Finding optimal solutions to cooperative pathfinding problems. In AAAI
4. M-star: Wagner, G., Choset, H.: Subdimensional expansion for multi-robot path planning. Artificial Intelligence 219, 1–24 (2015) 这里有很多参考文章(Why M-star is complete and optimal ?)
5. Bnaya, Z., and Felner, A. 2014. Conflict-oriented windowed hierarchical cooperative A\*. In (ICRA). Online Algorithm, ignore

CBS/ICTS Based Method:

1. CBS: G. Sharon, R. Stern, A. Felner, N. Sturtevant, Conflict-based search for optimal multi-agent path finding, in: Proceedings of the AAAI Conference on Artificial Intelligence, Toronto, Ontario, Canada, 2012
2. MA-CBS: Sharon, G., Stern, R., Felner, A., Sturtevant, N.R.: Conflict-based search for optimal multi-agent pathfinding. Artificial Intelligence 219, 40–66 (2015)
3. ICTS: Sharon, G., Stern, R., Goldenberg, M., Felner, A.: The increasing cost tree search for optimal multi-agent pathfinding. Artificial Intelligence 195, 470–495 (2013), ignore
4. Walker, T. T.; Sturtevant, N. R.; and Felner, A. 2018. Extended increasing cost tree search for non-unit cost domains. In International Joint Conference on Artificial Intelligence (IJCAI), 534–540. ignore
5. ECBS: Barer, M., Sharon, G., Stern, R., Felner, A.: Suboptimal variants of the conflictbased search algorithm for the multi-agent pathfinding problem. In: Symposium on Combinatorial Search (SoCS) (2014)
6. ICSB: Boyarski, E., Felner, A., Stern, R., Sharon, G., Tolpin, D., Betzalel, O., Shimony, E.: ICBS: improved conflict-based search algorithm for multi-agent pathfinding. In: International Joint Conference on Artificial Intelligence (IJCAI) (2015)
7. HCBS: Felner, A., Li, J., Boyarski, E., Ma, H., Cohen, L., Kumar, T.S., Koenig, S.: Adding heuristics to conflict-based search for multi-agent path finding. In: International Conference on Automated Planning and Scheduling (ICAPS) (2018)
8. Bypass+CBS: E. Boyarski, A. Felner, G. Sharon, and R. Stern. Don’t split, try to work it out : Bypassing conflicts in multi-agent pathfinding. In ICAPS-2015, Jerusalem, Israel, June 7-11, 2015.
9. Disjoint Splitting for Multi-Agent Path Finding with Conflict-Based Search, Jiaoyang Li, ICAPS 2019)

Prioritized Based Method:

1. H. Ma, D. Harabor, P. Stuckey, J. Li and S. Koenig. Searching with Consistent Prioritization for Multi-Agent Path Finding. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), pages (in print), 2019

3D Pipe：

1. From Multi-Agent Pathfinding to 3D Pipe Routing, Gleb Belov, SoCS 2020：
   1. Priority conflict-based search + A-star
   2. Priority planning + A-star
2. Position Paper: From Multi-Agent Pathfinding to Pipe Routing, Gleb Belov, 2019:
   1. ECBS + focal A-star

Any Angle Path:

1. K. Daniel, A. Nash, S. Koenig and A. Felner.  [Theta\*: Any-Angle Path Planning on Grids](https://web.archive.org/web/20160528140046/http://idm-lab.org/bib/abstracts/Koen10r.html). Journal of Artificial Intelligence Research, 39, 533-579, 2010. (supplement)
2. A. Nash, K. Daniel, S. Koenig and A. Felner. [Theta\*: Any-Angle Path Planning on Grids](https://web.archive.org/web/20160528140046/http://idm-lab.org/bib/abstracts/Koen07f.html). In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), 1177-1183, 2007:

Reference Link: [https://web.archive.org/web/20190717211246/http://aigamedev.com/open/tutorials/theta-star-any-angle-paths/](https://web.archive.org/web/20190717211246/http:/aigamedev.com/open/tutorials/theta-star-any-angle-paths/)

1. Optimal Any-Angle Pathfinding In Practice. Daniel Harabor Journal of Artifificial Intelligence Research 56 (2016). (Anya algorithm):

Reference Link:

<https://github.com/370417/symmetric-shadowcasting>

<https://www.albertford.com/shadowcasting/#is_blocking>

1. Compromise-free Pathfinding on a Navigation Mesh. Michael L. Cui.Joint Conference on Artificial Intelligence (IJCAI-17) (Polyanya , variant of Anya)

Non Holonomic Search:

1. Practical Search Techniques in Path Planning for Autonomous Driving. Dmitri Dolgov. 2008 (Hybrid A-star) Very Good Paper
2. Path Planning in Unstructured Environments: A Real-time Hybrid A\* Implementation for Fast and Deterministic Path Generation for the KTH Research Concept Vehicle. Karl Kurzer Karlsruhe Institute of Technology 2016 (Hybrid A-star supplement)
3. L. Wen, Y. Liu and H. Li, CL-MAPF: Multi-Agent Path Finding for Car-Like robots with kinematic and spatiotemporal constraints, Robotics and Autonomous Systems, 2021
4. Honig, W.; Kumar, T.; Cohen, L.; Ma, H.; Xu, H.; Ayanian, N.; and Koenig, S. 2017. Summary: multi-agent path finding with kinematic constraints. In International Joint Conference on Artificial Intelligence (IJCAI), 4869–4873 (MAPF-POST) ignore
5. Walker, T. T.; Chan, D.; and Sturtevant, N. R. 2017. Using hierarchical constraints to avoid conflicts in multi-agent path finding. In International Conference on Automated Planning and Scheduling (ICAPS). interesting but may be far from optimal without any bound
6. Gregor, Drivable path planning using hybrid search algorithm based on E∗and Bernstein-Bézier motion primitives. IEEE. (Hybrid+E-star)写得很好，但我不是看得很懂
7. A novel path planning methodology for automated valet parking based on directional graph search and geometry curve. Zhaobo Qin. State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University, Changsha. (Jump Point Search as direction heuristic + hybrid A-star)
8. Gunawan, S.A.; Pratama, G.N.; Cahyadi, A.I.; Winduratna, B.; Yuwono, Y.C.; Wahyunggoro, O. Smoothed A-star Algorithm for Nonholonomic Mobile Robot Path Planning. In Proceedings of the 2019 International Conference on Information and Communications Technology (ICOIACT), Yogyakarta, Indonesia, 24–25 July 2019; pp. 654–658. 找不到这篇论文的免费版
9. An Efficient and Robust Improved A\* Algorithm for Path Planning. Huanwei Wang. State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou. 2021. ignore 提供了其他论文参考
10. The EBS-A\* algorithm: An improved A\* algorithm for path planning. Huanwei Wang. State Key Laboratory of Mathematical Engineering and Advanced Computing, Zhengzhou. 2021. ignore

Larger Agent:

1. Li, J.; Surynek, P.; Felner, A.; Ma., H.; Kumar, T. K. S.; and Koenig, S. 2019. Multi-agent path finding for large agents. In AAAI Conference on Artificial Intelligence

Structure:

1. J. Li, Z. Chen, D. Harabor, P. Stuckey and S. Koenig. Anytime Multi-Agent Path Finding via Large Neighborhood Search. In International Joint Conference on Artificial Intelligence (IJCAI)
2. Code Reference:
3. <https://github.com/whoenig/libMultiRobotPlanning>
4. <https://github.com/Jiaoyang-Li/PBS>
5. Resource Attention：
   1. Andreychuk, A., Yakovlev, K.: Two techniques that enhance the performance of multi-robot prioritized path planning. In: International Conference on Autonomous Agents and Multi Agent Systems (AAMAS). pp. 2177–2179 (2018)
   2. EECBS: A Bounded-Suboptimal Search for Multi-Agent Path Finding
   3. HCBS + RTC: Improved Heuristics for Multi-Agent Path Finding with Conflict-Based Search
   4. Dynamic A\* and Lifelong Planning A\* may be useful

记录：

2023-03-06：

这段时间可以完成low-level的搜索方法与high-level规划方法的确定就很好了，第二个比较麻烦的方面就是要找到合适的参考代码并进行移植。对于传统的搜索方法的改进主要体现在：（1）启发式（2）限制branch数目（3）independent detection（4）创建window移动窗体（4）Operator decomposition （5）Conflict Avoid Table

2023-03-08：

与多智体路径规划不同的是，管道规划中行走过的路径不允许再被占据，所以Prioritized Planning 才会比较占主导，不过CBS应该仍然是最优结果的求解器。但我估计在三维上使用CBS会比在二维上开销大得多。

1. M-star似乎不适合于管道路径问题，因为路径不允许二次占据

2023-03-09:

总结来说，与常规的多智慧体路径规划相对比，主要是agent merge的操作失效了，（1） Operator Decomposition（单纯作为branch发散的问题方案使用）（2）ByPass方法 （3）启发式 这部分仍然是有效的。我的计划：

1. 完成两个求解器：
   1. 基于CBS + Search 的求解器 (CBS+A\*, ECBS)
   2. 基于 Prioritized Planning 的求解器 (PBS)
2. 我需要对比一下人工设计与算法设计的纯路径的差异
3. 路径平滑还是一个大问题，我预估这是一个Motion Planning Problem，我可能需要参考一下Hybrid A\* 或 基于simulation 的方法（例如Model Predictive Control，RRT\*）

2023-03-10:

A\*-star的变种应该是有用的：A-star（wiki）

2023-03-22：

目前完成了简单的Space-Time Astar与CBS的基础。在确认了方向无误后，先考虑完成一个完整的应用，先在小的测试环境上完成对（1）不同尺寸（2）平滑路径（3）合并与交集，三个问题的处理。

我认为我之前参考multi-agent path finding是有一定的偏差的，因为agent path finding的冲突需要考虑运动时间，但管道不需要考虑运动时间，因此可能Any-Angel Path finding与CBS混合是一个更合适的选择。

2023-03-23：

早期的两个思路是：（1）先用Astar搜索最优路径，然后做路径短接后处理，再在转角位使用特定的平滑方法。（2）直接在非完整的连续空间进行搜索

目前的三个问题主要是：（1）管道尺寸不一致，（2）平滑可能产生干涉或密集区域无法平滑

1. 关于LA-MAPF的问题，我认为如果使用disjoint splitting的方法则约束为一整个球体，否则用常规方式，则使用MC-CBS的方法，用交集约束

2023-03-24：

需求比我预想中多，而且由于我对目前的情况理解太少，似乎很多的需求缺乏启发式，它们的衍生是非线性的，那意味着无法在一开始的时候就知悉。这样迭代式的开发可能比想象中重要。

这个项目使用连续搜索比离散搜索更合适。MAPF-LNS有迭代形式的思想，我需要参考一下。

那些流体参数可能可以塞进路径的梯度优化里面。

可以通过全局梯度下降来平滑曲线。

2023-03-25：

既然找不到参考，要不直接看RRT的路径规划文章

或者先弱化一下graph

2023-03-27：

我介意的问题在于：

1. 特殊的约束是设计前知道，设计中知道，还是设计后知道
2. 特殊的约束是设计中一直保持，还是设计中存在
3. 约束是不是软约束，即是同一约束不同选择有不同值
4. 进口与出口一定要与截面正交（希望如此）

我没太多的把握，第1，这是个NP-Hard难题，这意味着必须有人的启发式，否则无法解决问题。第2，这几乎一定是个迭代求解问题，并且需要约束启发式的选择可能。第3，既然是个迭代问题，应该尽量使求解快速。

角度的两条限制式：

杜宾启发式在三维上完备，但怎么确定最短？（这些曲线圆滑限制太大，我不想用），我希望可以直接使用梯度优化确定最优曲线。

我需要方法来bound住路径

2023-03-28:

1. CBS + Reactive Path finding（例如CBS + Voronoi 势能场，可能陷入局部点无法逃逸）

CBS + Reactive Path finding + hybrid method可能是个好选择，其实我主要执着于A\*的branch问题上，我希望有替代的方式

杜宾曲线是不是转弯半径越小则结果路径越小呢？