Literature Review

Joohyun, Park
2018..09.14.(Fri)

Path Planning Combining Improved Rapidly-Exploring Random Trees
with Dynamic Window Approach in ROS
2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)

요약 - RRT(Rapidly-exploring Random Trees)와 DWA(Dynamic Window Approach)를
결합해Global Path planning과 local Path planning을 나눠서 수행.

---Previous methods / global planning
'PRMs', 'RRT' use priori information of obstacles and generate trajectories between the initial position and the global position.
-> static world, do not take sensory information into consideration.

---The later planning methods / local planning
'APF', 'VFH+', 'ORM', 'FGM' compute appropriate robot heading angle for obstacles free navigation.
-> reactive path planning
-> not sufficient for taking the robot dynamic.
-> these approaches efficiently genearate direction outputs.
'CVM', 'VO', 'DWA' use dynamic properties of robots in order to perform obstacle free navigation.
-> presume the robot travels as arcs.

---Global and local path planning problem
was combined into a single search using a combined 2-D and higher dimensional state-space

A star, Dijkstra algorithm offer optimal path by searching grid map, efficiency cannot be ensured when encountering large-scale environment.

--> Biased RRT algorithm is integrated as a global planner and DWA a local planner.
RRt can provide an efficient path in large scale and DWA can straightway give proper velocity. RRT is a global path planner, DWA computes translational velocity and rotational velocity for the robot.
-> this get a soomther path with less time(27% quicker than Astar) and higher speed(8% faster than Astar).
-> weakness : 2-D grid map. real world is rarely existing.
there are not any dynamic obstacle in this simulations.

UAV path planning based on Improved Rapidly-exploring Random Tree
2018 Chinese Control And Decision Conference (CCDC)

요약 - the efficiency and success rate of the basic RRT algorithm for path planning need to be improved.
- this algorithm combined bi-RRT and dynamic P value, step length that adjusted.

path planning methods
1. ant colony algorithm
2. artificial potential field -> Astar Algorithm
3. RRT

RRT algorithm lead the search to black region by random sampling of the state space.
RRT algorithm can more quickly than Astar algorithm is also easy to apply to search for high-dimensional search and dynamic obstacle environment.

RRT does not consider the comprehensive cost of the path in the search process.
Node extension leads to the inherent randomness of the planned path -> Not optiam sol.

Proposed Method
1. Use dynamic P value
-> the P value should be reduced in order to increase the randomness of the search.
-> Selecting X_near
2. Determin dynamic step length
-> the length of step has an important influence on the number of nodes and the success rate of path planning.
--> using the dynamic step can achieve a good balance between efficiency and success rate.

So step length can be dynamically adjusted according to angle and constraint rules, and combined bidirectional RRT and dynamic P value.

RRT의 노드 개수는 줄었지만 그래도 많은 계산량을 요구함. -> Not real time
Computation time 줄었지만 너무 미미하게 줄어듬.
obstacle 개수가 늘어나면 계산시간은 더 걸림. 여기에서는 100개를 추가했는데 기존 알고리즘보다 1.2sec 느려짐.

실험자체도 컴퓨터 시뮬레이션으로만 진행되었고, 실제 비행 환경은 더 복잡하고 변화무쌍하므로 더 확인하고 수정해야한다고 함.

Global and Local Path Planning Study in a ROS-Based Research
Platform for Autonomous Vehicles
Journal of Advanced Transportation
Volume 2018, Article ID 6392697, 10 pages

- the aim of this work is to integrate and analyze the performance of a path planning method based on Time Elastic Bands(TEB) in real research platform based on Ackermann model.

-> The study is done by analyzing the trajectory generated from global and local planners.

Navigation module is divided into a global planner and local planner.
-> the first one finds the optimal path with a prior knowledge of the environment and static obstacles.
-> the second one recalculates the path to avoid dynamic obstacles.

Path Planning.
- 2D space of a single plane x-y where the z component is neglected.
- global planner : uses a priori information of the environment to create the best possible path.
  -> using like Astar, Dijkstra, Silhouette, RRT algorithms.
    -> 높은 computing power을 요구함.
  -> Dijkstra를 이용했는데 이유는 간단한 디버깅, 코드의 단순함.
    -> grid cell map. -> 연산량 많은 단점.
- local planner : recalculates the initial plan to avoid possible dynamic obstacles.
  -> the local planner creates new waypoints taking into consideration the dynamic obstacles and the vehicle constraints.
    -> TEB which consists of deforming the initial global plan considering the kinematic model of the vehicle and updating the local path based on dynamic obstacles or the possible deviation from the path.
      -> low computational cost.

This algorithm is useful in both of these events and when a dynamic obstacles is in front of the vehicle because of the continuous update of the local path.

The TEB local planner generates the commands of velocities and steering angles that allow the vehicle to reach the goal.

- 2-D에서 실험했고, Z축을 제외해서 path planning을 했기 때문에 Z축에 대한 정보는 이용하지 않음.
- 실험결과가 매우 좋음.
- Dijkstra를 이용해 연산량이 많음.

Survey of Robot 3D Path Planning Algorithms
Journal of Control Science and Engineering archive
Volume 2016, March 2016
Article No. 5
Hindawi Publishing Corp. New York, NY, United States

-This paper classifies all the methods into five categories based on their exploring mechanisms and proposes a category.

3D path planning algorithms
- Sampling based algorithms
- Node based optimal algorithms
- Mathematic model based algorithms
- Bioinspired algorithms
- Multifusion based algorithms

1. Sampling based algorithms -> Not real time / search randomly to achieve a feasible path
- this kind of method need some preknown information of the whole workspace.

●Active
- RRT, DDRT, RRTstar
- Artificial Potential Field

●Passive
- 3D Voronoi, RRG
  -> it forms a 3D obstacle free network based on preknown knowledge of the whole envir.
- PRM, k-PRM, s-PRM
  -> 3D에서는 좋은 성능을 보임. 단, 연산량이 매우 많음.

2. Node based Optimal Algorithms
- these algorithms explore through the decomposed graph.
- however, they do not collide with each other and just express the same thing
  where algorithms like this kind deal with discrete optimization
  based on grid decomposition.
  -> grid decomposition..

3. Mathematic Model based algorithms
- based algorithms include linear algorithms and optimal control.
- general mathematic model based algorithms often fail.

4. Bioinspired algorithms
- this kind of planning methods leaves out the process of constructing complex environment models to search a near optimal path based on stochastic approaches; it overcome the weakness where general mathematic model based algorithms often fail.

Sampling based algorithms

- RRT
  - can find a path to the goal, it is still the problem that RRT explores based on Monte Carlo random sampling, which is always biasing explored region as it will increase with the time.
    -> will consume much time to find a way out when the environments are clutterd, let alone converge to the optimal.
    -> 많은 시간 씀.

-RRG
  - it generates a complex network like PRM and thus cannot find the optimal path by itself.

-RRT star
  - the whole time consumption increases, and it cannot work to generate multipath.

-PRM series
  - the expanding of the exploring graph, the expense on collision checking increase.
  - Hsu et al. -> solved the 'dynamic threat work' problem of PRM by introducing a concept of "milestone", that is, (state x time) which creates a real time graph connecting the initial point and the goal point.
    -> time을 사용해서 해결함.

- Voronoi
  - static obstacles

Node based algorithms

- Dijkstra
  - relies much on the priority queue Q data structure type, which influences the total exploring time.

- Astar, THETAstar
  - it consumes much time to check unexpected neighbors.
  - THETAstar reduces the searching compared to Astar.

Conclusion

1. sampling based algorithms
   - the random initial guess ensures escaping of local minimum, and this kind of algorithms does not rely much on environmental representation.
   - active algorithms can find the optimal path by their own. passive algorithms cannot
   - these approaches are appropriate for on-line implementation.
     -> they have a high time efficiency, with the ability to handle dynamic and static threat

2. Node based optimal algorithms
    - they originate from dynamic programming approaches and thus cannot further optimize the result beyond the decomposition of the environment
    - The results of this kind of algorithms rely much on the preconstructed graph and can be combined with other methods to achieve global optimal.

3. Mathematic model based algorithms
    - this kind of methods loads a heavy computational burden on computer

4. bioinspired algorithms
    - this kind of algorithms is suggested to work off-line, even though they can handle dynamic threats.

5. Multifusion based algorithms
    - are designed to work real time, with strong environmental adaption.
    - merging the merits of several algorithms
    - EMA (Embedded Multifusion Algorithms)
      -> Voronoi, neural network potential field, RDR, VVP, hybrid bioinspired
    - RMA (Ranked --)
      -> PRM node based, GIS-MCDA, visibility graph node based, sampling based