A Mixed Local-Global Solution to Motion Planning within 3-D Environments

Ryan Hurley∗ and Rick Lind†
University of Florida

Joseph J. Kehoe‡
Scientific Systems Company, Inc.

Autonomous flight through urban environments requires methods to generate trajectories that traverse a region and its associated obstacles. This paper introduces the development of a 3-dimensional motion planning algorithm using a random dense tree whose branches are motion primitives from a 3-dimensional version of the Dubins car called the Dubins airplane. The motion primitives consist of 3-dimensional maneuvers formulated as combinations of turn segments and straight segments with an associated constant rate of climb. The resulting motion planner builds the tree by pruning nodes that intersect 3-dimensional obstacles while connecting the remaining nodes with the motion primitives. An example demonstrates the motion planner can avoid building-style obstacles and even bridges using feasible paths that are sub-optimal solutions to minimize the cost of flight time.

I. Introduction

The maturation of micro air vehicles has introduced a class of aircraft of appropriate size and airspeed to enable flight through urban environments. Such flight will require immersion amongst obstacles that will require fully 3-dimensional maneuvering in a densely-cluttered space. Certainly maximizing agility will be critical to mission performance; however, motion planning will also be critical to taking advantage of any flight capabilities.

Such close-proximity flight presents challenges for path planning. In particular, significant portions of the environment extending beyond obstacles may be restricted because the vehicle may not have sufficient agility to maneuver safely. As such, the classic approach of planning that considers waypoints may not be suitable unless some guarantee of feasible maneuvering is provided between those waypoints.

Inclusion of dynamically-feasible motions in a planned trajectory is typically treated in either a direct or a decoupled fashion. Direct planning methods, such as optimal control, consider a representation of the vehicle dynamics in the formulation of the planning problem and directly solve for optimal system inputs. Alternatively, indirect methods use a simplified model of vehicle motion to plan a reference path and then smooth the path to satisfy dynamics using methods such as feedback control. Direct methods compute optimal trajectories but are often intractable for realistic problem descriptions whereas indirect methods often exhibit tractable complexity properties that come at the expense of optimality.

∗Graduate Student, fgdonpat@ufl.edu
†Associate Professor, ricklind@ufl.edu
‡Senior Systems Engineer, joseph.kehoe@ssci.com
A variety of techniques manipulate this tradeoff by directly including dynamics in the planning process. For example, systems which exhibit differential-flatness properties admit solutions that can be represented parametrically in terms of a set of flat outputs and their derivatives.\textsuperscript{2–4} Another approach uses mixed-integer linear programming to model dynamic constraints as a set of switching bounds on system velocities and accelerations.\textsuperscript{5–7} A planning technique is introduced that utilizes a sampled-dynamics model which employs a set of dynamically-consistent motion primitives.\textsuperscript{8, 9} Additionally, recent advances in randomized planning allow the use of any of these techniques as local trajectory generation methods for growing a probabilistic tree of actions to explore the solution space.\textsuperscript{10–12}

The concept of basic maneuvers, or motion primitives, is central to several of these investigations into feasible-path motion planning. A critical foundation was established by Dubins for a 2-dimensional car.\textsuperscript{13} The concept of a Dubins car provides a closed-form solution for optimal trajectories and has been used for many types of planning such as the traveling-salesman problem.\textsuperscript{14} This foundation is used for several studies into aircraft motion but the process limits that motion to a 2-D plane.\textsuperscript{15–18} One approach expands the original 2-D formulation into a 3-D framework but does not deal with constraints in the climb rate or specific values of these climb rates as is the case with motion primitives.\textsuperscript{19} Complete analogues to the Dubins car in 3-D are being developed to account for the shortest path between two points with associated heading constraints\textsuperscript{20} and with associated heading and flight-path angle constraints.\textsuperscript{21}

This paper introduces an approach for motion planning of 3-D trajectories amongst close-proximity obstacles; specifically, a method that combines random dense trees with motion primitives is developed for fully 3-D environments. The procedure considers expanding from the initial configuration along feasible trajectories, which are the motion primitives, to an expanding set of nodes that are chosen based on tree growth. These feasible trajectories are optimal between the nodes although the resulting trajectory has no guarantees of optimality for the entire path from the initial configuration to the final configuration.

The approach is demonstrated for efficient generation of trajectories through a representative urban environment. The tree growth is constrained to consider only future nodes that are limited in distance from the current nodes along with ensure a feasible trajectory can connect those nodes without intersecting any obstacles. This constraining has the effect of adding additional nodes to the tree and to increase the probability of finding a feasible solution among dense obstacles.

\section*{II. Motion Primitives}

\subsection*{A. Three-Dimensional Dubins Airplane}

A family of motion primitives is generated using a Dubins airplane. This concept arises by extending the family of 2-D trajectories associated with the classic Dubins car\textsuperscript{13} into 3-D trajectories.\textsuperscript{19, 20} The basic concept considers motion that can be straight or turning with altitude that can remain constant or change due to diving or climbing.

The model of kinematic motion operates in a configuration space of \( \mathcal{C}^4 \) spanned by three Euclidean position variables, \((p_x, p_y, p_z) \in \mathcal{R}^3\), and an angle of \( \psi \in \mathcal{R} \) that describes heading. The vehicle is constrained during motion to having a constant velocity of \( V > 0 \in \mathcal{R} \), a constant turn rate of \( \omega \in \mathcal{R} \), and a constant rate of change of altitude given as \( \gamma \in \mathcal{R} \). The resulting states are governed by Equation 1.
The trajectories associated with the model in Equation 1 represent maneuvers that the vehicle can perform. Since most vehicles can vary their rates of change, it is reasonable to define a set of parameters such that $\Omega = \{\omega_1, ..., \omega_n\}$ represents the set of possible turn rates and $\Gamma = \{\gamma_1, ..., \gamma_m\}$ represents the set of possible climb rates.

A set of motion primitives are then defined that represent all possible maneuvers. Each element in this set is actually a trajectory defined by the time-varying values of position and orientation during the maneuver. As such, any motion primitive, $X \in \mathcal{X}$, is parametrized by the duration of the maneuver, $\tau > 0$, along with turn rate, $\omega$, and climb rate, $\gamma$, as given in Equation 2.

$$\mathcal{X} = \left\{X(\tau, \omega, \gamma) : X = \int_0^\tau \begin{bmatrix} V \cos \psi \\ V \sin \psi \\ \gamma \\ \omega \end{bmatrix} dt, \omega \in \Omega, \gamma \in \Gamma \right\}$$ (2)

A critical feature of this set is the notion of feasibility. Essentially, any member, $X \in \mathcal{X}$, is constrained so the evolution of the trajectory is constrained by the differential relationship in Equation 1. The resulting set is a collection, or library, of feasible maneuvers that can be achieved by the vehicle.

A motion primitive, $X(\tau_2 - \tau_1, \omega, \gamma) \in \mathcal{X}$, represents a trajectory of the vehicle with a turn rate of $\omega \in \Omega$ and a climb rate of $\gamma \in \Gamma$ that lasts for a duration of $\tau_2 - \tau_1$. A vehicle starting at an initial configuration of $C(\tau_1) \in C^4$ that undergoes this motion primitive will reach a final configuration of $C(\tau_2) \in C^4$ as shown in Equation 3.

$$C(\tau_1) + X(\tau_2 - \tau_1, \omega, \gamma) = C(\tau_2)$$ (3)

B. Motion Planning

The motion primitives are suitable for motion planning to connect two configurations in $C^4$. In particular, sequences of motion primitives are effective at representing complicated trajectories that connect a wide range of configurations. The choosing of such motion primitives represents a type of motion planning.

The original formulation of the Dubins car, which is a subset of the Dubins airplane, has interesting properties for motion planning. This Dubins car actually operates in a configuration space of $C^3$ composed of 2 positions and a heading angle. A pair of results are known for the Dubins car when considering trajectories through an environment without obstacles.

- Any position in $R^2$ that lies outside the turn radius of the vehicle can be reached using a 2-primitive sequence from any initial configuration in $C^3$. These sequences are composed of a turn maneuver followed by a straight motion given as either a left-straight sequence or a right-straight sequence. Note that a 2-primitive sequence is only able to connect any configuration with any position but can not guarantee a desired value for the final heading.
• Any configuration in $C^3$ can be reached using a 3-primitive sequence from any initial configuration in $C^3$. Most importantly, a closed-form solution for the optimal trajectory is derived.$^{13}$ This optimal trajectory is composed of either turn-straight-turn sequences or turn-turn-turn sequences involving a specific sequence of turns being right or left.

The Dubins airplane is used for motion planning by adopting some results from the Dubins car. The motion primitives for the airplane actually describe the same trajectory as the motion primitives for the car except that vertical translation results by adding a climb rate. In this sense, the motion in the horizontal 2-D plane of $x-y$ states is identical for the airplane and car. Some results for trajectories that connect configurations in the absence of obstacles are immediately realized by utilizing the similarity in the horizontal plane.

• A set of positions in $R^3$ can be reached using a 2-primitive sequence from any initial configuration in $C^4$. These sequences are composed of a turn maneuver followed by a straight motion with a constant rate of altitude change, either climb or dive, included during the duration. This sequence follows a 2-primitive sequence for the 2-D Dubins car to travel the horizontal distance along with the constant rate of climb or dive active during the duration of each primitive to traverse the vertical distance. Note that the 2-primitive sequence is again not able to guarantee a desired value for the final heading.

• A set of configurations in $C^4$ can be reached using an optimal 3-primitive sequence from any initial configuration in $C^4$. The actual sequence is the optimal 3-primitive sequence from the closed-form solution of the Dubins car with an additional change in altitude being included throughout the duration of the sequence. The set of turn-straight-turn sequences connecting a pair of example configurations is shown in Figure 1 along with the optimal path as computed from the closed-form solution.

This approach is limited to configurations in which the time to travel horizontally is greater than the time to travel vertically. Essentially, the vehicle travels to a horizontal location by assuming the vertical distance is traveled along the way. Such an approach does not generate feasible solutions for all configurations that require more time for vertical translation than horizontal translation; however, techniques are developed that introduce waypoints to optimize the path and allow sufficient time to increase vertical translation.$^{24}$ Even so, the current constraint is actually not restrictive since a set of intermediate configurations are chosen to satisfy the assumptions on time to travel using the approach for motion planning in this paper.

### III. Random Dense Trees

Randomized methods for path planning are formulated to consider systems with complicated dynamics. The fundamental feature of such methods is a localized approach that considers sequentially expanding into a search space to rapidly and efficiently find sub-optimal solutions. A variety of methods, including probabilistic roadmaps and random dense trees (RDTs), are developed; however, the use of random dense trees is adopted in this paper due to its ability to directly handle motion primitives and generate feasible trajectories for models of realistic vehicles.$^{25–30}$

A tree of particular interest is the rapidly-exploring random tree that is formulated specifically to consider dynamics and differential constraints.$^{12,27}$ The algorithm biases tree growth toward unexplored areas of the space and hence focuses on rapid exploration.

The growth of this type of tree is summarized as a procedure of node selection and expansion. The step for node selection is initiated with a sampled configuration that is chosen from a uniform distribution of the configuration space. A distance metric is then used to determine the closest point in the existing tree. During the expansion step, the selected node is extended incrementally.
Figure 1. Possible (—) and optimal ( ——) Turn-Straight-Turn 3-D Dubins airplane solutions.

toward the sampled configuration using a local planning method. This incremental extension can be performed using several approaches including a step size of fixed magnitude, a step size proportional to the distance from the current configuration, or a random selection without constraints.

The expansion process is depicted in Figure 2 for a generalized tree in a configuration space of \( C^n \). The tree is grown from an initial node, \( N_0 \in C^n \), and extends into an environment containing obstacles. The initial step of node selection has \( N_{rand} \in C^n \) chosen from a random sampling along with the nearest node in the existing tree as \( N_{near} \in C^n \). The expansion step builds a branch from \( N_{near} \) toward \( N_{rand} \) along the trajectory connecting the two configurations. Finally, a new node of \( N_{new} \in C^n \) is added at the end of the new branch. The algorithm proceeds in this fashion until a branch of the tree reaches the goal within some specified tolerance.

Figure 2. RRT algorithm. A) Sampling step. B) Expansion step.
A tree can be grown into any $\mathcal{C}^n$ space. The set of nodes are chosen by sampling that space to find appropriate configurations. Branches that provide transformations from $\mathcal{C}^n \rightarrow \mathcal{C}^n$ provide the connectivity between these configurations. A representative tree is shown in Figure 3 that shows growth into a configuration space of $\mathcal{C}^3 = \mathcal{R}^3$ composed of 3 Euclidean positions.

![Figure 3. Growth of 3-D Tree after 100 Iterations (top left), 250 Iterations (top right) and 500 Iterations (bottom)](image)

**IV. Motion Planning**

An approach for motion planning is formulated that combines 3-D random dense trees with 3-D motion primitives, specifically, the approach uses the tree to expand nodes into the environment and then use trajectories from motion primitives as branches that connect the nodes. This approach accounts for obstacles using a pruning algorithm until the vehicle reaches a node from which an optimal 3-primitive sequence will connect to the final configuration. The initial configuration is given as $C_{initial} \in \mathcal{C}^4$ and the final configuration is given as $C_{final} \in \mathcal{C}^4$.

1. **Select a Node:** A point, $C_{i+1} \in \mathcal{C}^4$, is selected from the subspace of the feasibility space which is spanned by the position variables. This node is considered an extension beyond the closest node, $C_i \in \mathcal{C}^4$, of the current tree as determined by a distance metric.

2. **Extend a Branch:** A branch is generated to connect the current configuration, $C_i$, with the next node, $C_{i+1}$, in the tree. This branch is generated using a 2-primitive sequence composed of a turn maneuver followed by a straight motion. The actual primitives used in the sequence...
are chosen to have the minimum sum of durations subject to the constraint that they must connect the configurations after that time as shown in Equation 4.

\[
\min \tau_1 + \tau_2 \quad X_i \in \mathcal{X} \\
X_s \in \mathcal{X}
\]

subject to \( C_{i+1}(t + \tau_1 + \tau_2) = C_i(t) + X_t(\tau_1, \omega, \gamma) + X_s(\tau_2, 0, \gamma) \)

3. **Obstacle Avoidance**: A pruning method is used to ensure obstacle avoidance. This method does not directly consider the location of the obstacles to optimize tree growth; rather, it simply prunes nodes and branches that lie within an obstacle. The node selection thus remains random with some of the nodes being eliminated by a check on the node location and the obstacle locations.

This pruning notes that a set of locations, \( \mathcal{O} \), may be defined that encompasses the obstacles. The definition in Equation 5 uses a simple orthogonal polyhedron approximation such that each obstacle has limits on east range, \([x_1, x_2]\), north range, \([y_1, y_2]\), and altitude range, \([z_1, z_2]\), for \( k \) obstacles.

\[
\mathcal{O} = \left\{ O : O = \begin{cases} 
  x \in [x_1, x_2] \\
  y \in [y_1, y_2] \\
  z \in [z_1, z_2] 
\end{cases} \quad \forall i \in [1,k] \right\}
\]

The growth of the tree occurs such that a node, \( C_{i+1}(t + \tau_1 + \tau_2) \), as in Equation 4 is valid if neither that node nor a path to that node intersect any obstacles as described in Equation 6:

\[
C_{i+1} \text{ is a valid node if} \begin{cases} 
  C_{i+1}(t + \tau_1 + \tau_2) \notin \mathcal{O} \\
  \exists X_t(\tau, \omega, \gamma) \in \mathcal{X} \text{ with } X \notin \mathcal{O} \quad \forall \tau \in [t, t + \tau_1] \\
  \exists X_s(\tau, 0, \gamma) \in \mathcal{X} \text{ with } X \notin \mathcal{O} \quad \forall \tau \in [t + \tau_1, t + \tau_1 + \tau_2] 
\end{cases}
\]

4. **Check for Solutions**: The final branch of a tree is determined as the closed-form solution for the Dubins car with altitude variation that connects two configurations in \( \mathcal{C}^4 \). This 3-primitive solution represents the optimal trajectory with minimum time to travel between those configurations so it is assumed that additional nodes, and their associated sub-optimal 2-primitive solutions, will only increase the total cost of the motion planning. The expansion process evaluates if such an optimal 3-primitive sequence exists between every tree node and the final configuration that does not intersect any obstacles as in Equation 7:

\[
C_{i+2} = C_{\text{final}}
\]

if \( \exists \) optimal \( X_1, X_2, X_3 \in \mathcal{X} \) such that

\[
C_{\text{final}}(t + \tau_1 + \tau_2 + \tau_3) = C_{i+1}(t) + X_1(\tau_1, \omega_1, \gamma_1) + X_2(\tau_2, \omega_2, \gamma_2) + X_3(\tau_3, \omega_3, \gamma_3) \\
X_1(\tau_1, \omega_1, \gamma_1) \notin \mathcal{O} \quad \forall \tau \in [t, t + \tau_1] \\
X_2(\tau, \omega_2, \gamma_2) \notin \mathcal{O} \quad \forall \tau \in [t + \tau_1, t + \tau_1 + \tau_2] \\
X_3(\tau, \omega_3, \gamma_3) \notin \mathcal{O} \quad \forall \tau \in [t + \tau_1 + \tau_2, t + \tau_1 + \tau_2 + \tau_3]
A critical aspect of the tree expansion is using only 2-primitive sequences to connect nodes of $C_i$ to $C_{i+1}$. It is known that such sequences are not able to guarantee a desired heading at $C_{i+1}$; however, the heading is actually only constrained at the initial configuration and the final configuration. The nodes of tree are simply intermediate nodes so they are not associated with any heading constraint. As such, a 2-primitive sequence is computationally faster to generate for node connection.

Also, the choice of node selection is a critical aspect that provides several benefits. The distance between nodes is kept relatively small with respect to the horizontal distance and even smaller with respect to the vertical distance. This limitation ensures that the time to travel the horizontal distance is greater than the time to travel the vertical distance which is assumed for this approach of motion planning using primitives. The close spacing of nodes also allows for complicated trajectories that are composed of many primitives to ensure the path is able to avoid dense obstacles.\textsuperscript{22,23}

V. Example

A. System

Motion planning is demonstrated for an aircraft that needs to traverse an urban environment. In this case, the vehicle has the dynamic properties given in Table 1 that are based on measurements from a class of micro air vehicles at the University of Florida. In this case, a distinct set of values for turn rate are chosen while the climb rate is allowed to vary as any value within a given range.

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>forward velocity</td>
<td>40 ft/s</td>
</tr>
<tr>
<td>turn rate</td>
<td>${0, \pm 15, \pm 20, \pm 30}$ deg/s</td>
</tr>
<tr>
<td>climb rate</td>
<td>$[-30, 30]$ ft/s</td>
</tr>
</tbody>
</table>

Table 1. Vehicle properties for examples.

The vehicle starts at a position of (0,0,0) and a heading of 45° while it is required to travel to a position of (500,500,200) and a heading of 90°. The path between these configurations must avoid obstacles consisting of 2 large towers, 1 small tower, a covered walkway, and an elevated bridge. The details of the obstacles are presented in Table 2.

<table>
<thead>
<tr>
<th>Obstacle</th>
<th>Coordinates of Center</th>
<th>dx</th>
<th>dy</th>
<th>dz</th>
</tr>
</thead>
<tbody>
<tr>
<td>North Tower</td>
<td>(50,250,100)</td>
<td>100-0</td>
<td>300-200</td>
<td>200-0</td>
</tr>
<tr>
<td>Covered Walkway</td>
<td>(175,275,25)</td>
<td>250-100</td>
<td>300-250</td>
<td>50-0</td>
</tr>
<tr>
<td>Northeast Tower</td>
<td>(275,275,100)</td>
<td>300-250</td>
<td>300-250</td>
<td>200-0</td>
</tr>
<tr>
<td>Elevated Bridge</td>
<td>(275,175,175)</td>
<td>300-250</td>
<td>250-100</td>
<td>200-150</td>
</tr>
<tr>
<td>East Tower</td>
<td>(250,50,100)</td>
<td>300-200</td>
<td>100-0</td>
<td>200-0</td>
</tr>
</tbody>
</table>

Table 2. Tower, walkway, and bridge dimensions and locations. All units in feet.

B. Motion Planning

A first tree is grown to compute a sub-optimal trajectory as a combination of 2-primitive sequences followed by an optimal 3-primitive sequence. A total of 9 solutions are identified with the lowest cost of travel time being 19.04 s. The vehicle initially travels east until sharply turning north while
climbing to avoid an obstacle. The last node in the tree is positioned between obstacles with a clear path to the final configuration. The complete tree with all nodes is shown in Figure 4 along with the lowest-cost trajectory.

A second tree is grown to evaluate a new set of solutions. This tree generates 5 solutions of which 18.39 s is the travel time for the most-favorable trajectory. This most-favorable trajectory, as shown in Figure 5, results from an easterly path that climbs and then dives due to tree growth followed by an optimal 3-primitive sequence to the final configuration.

A third tree is grown to generate a final set of trajectories. This tree generates another 5 solutions with 18.24 s being the shortest duration for a trajectory. This trajectory as shown in Figure 6 has the vehicle climbing while moving north-east through growth of the tree until an optimal 3-primitive sequence takes the vehicle over a walkway and past a building to reach the final configuration.
C. Evaluation

The trajectories resulting from the motion planning are sub-optimal in the sense that they are based on a set of locally-optimal sequences; however, the final 3-primitive sequence is actually a closed-form solution of optimal trajectory from the last node in the tree to the final configuration. As such, the trees are evaluated with particular attention given to the role of this final 3-primitive sequence in determining the cost of time to travel.

The characteristics of each trajectory are different because of the random nature of the tree growth. The specific parameters of each primitive used in the sequences are evaluated to determine their role in minimizing the cost of time to travel. In this case, the first tree and third tree have 3 sets of locally-optimal 2-primitive sequences followed by the optimal 3-primitive sequence whereas the second tree has 2 sets of locally-optimal 2-primitive sequences followed by the optimal 3-primitive sequence.

The flight path resulting from each tree is visibly different as a result of difference in the motion primitives; specifically, the turn rate and climb rate vary. The turn rate for each primitive in each tree is given in Table 3 and indicates the first pair of trees start with a low rate of turn and finish with a high rate of turn while the third tree starts with a high rate of turn and finishes with a low rate of turn. Similarly, the values in Table 4 indicate that the first pair of trees start with a higher climb rate than the third tree but then finish with a lower climb rate than the third tree. Note that the climb rate is held constant for an entire sequence of primitives.

<table>
<thead>
<tr>
<th>Example</th>
<th>Sequence-1</th>
<th>Sequence-2</th>
<th>Sequence-3</th>
<th>Sequence-4</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$X_t$</td>
<td>$X_s$</td>
<td>$X_t$</td>
<td>$X_s$</td>
</tr>
<tr>
<td>1</td>
<td>20</td>
<td>0</td>
<td>-30</td>
<td>30</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>0</td>
<td>30</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>-30</td>
<td>0</td>
<td>-30</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 3. Turn Rate for Motion Primitives

The distance traveled by each primitive varies due to these variations in turn rate and climb rate. The norm of the distance for each primitive is given in Table 5 to indicate that the first tree
Table 4. Climb Rates for Motion Primitives

<table>
<thead>
<tr>
<th>Example</th>
<th>Sequence-1</th>
<th>Sequence-2</th>
<th>Sequence-3</th>
<th>Sequence-4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4.24</td>
<td>-0.71</td>
<td>16.52</td>
<td>13.52</td>
</tr>
<tr>
<td>2</td>
<td>6.15</td>
<td>-5.23</td>
<td>0.00</td>
<td>19.92</td>
</tr>
<tr>
<td>3</td>
<td>3.31</td>
<td>-1.30</td>
<td>25.83</td>
<td>10.80</td>
</tr>
</tbody>
</table>

Table 5. Distances for Motion Primitives

The cost relating path optimality is travel time; consequently, the time to travel is a critical metric to evaluate for each primitive. These times are given in Table 6 and indicate the first tree has the longest duration while the third tree has the shortest duration. A decomposition of the travel times, similar to the travel distances, indicates that the second tree actually has the shortest duration of optimal 3-primitive sequence while the third tree has the longest duration of that final optimal sequence.

Table 6. Time Durations for Motion Primitives

The role of the final 3-primitive sequence is somewhat unexpected. It is logical to assume that the trajectory would have a lower cost if this optimal sequence comprised a larger portion; however, the converse happened for these trajectories. The trajectory should indeed be composed mostly, and eventually entirely, of the optimal 3-primitive sequence from the Dubins car as the environment is cleared of obstacles. The presence of obstacles changes the condition for optimality from simply maximizing the portion devoted to the optimal 3-primitive sequence to choosing a better place from which to start that optimal 3-primitive sequence. In other words, the trajectory will be costly if the tree meanders among the obstacles for too long before entering clear space even if that last node is close to the final configuration.
D. Statistics

A set of 500 trajectories are generated using this approach to motion planning. The random dense trees are each grown by expanding into the environment using unique nodes and branches. The duration times for each trajectory are collected into the histogram of Figure 7.

![Histogram of Trajectory Duration](image)

Figure 7. Histogram of Trajectory Duration

The histogram demonstrates several properties of motion planning using random dense trees. Certainly the variation in durations results from trees whose trajectories were local minima due to the sub-optimal nature of the planning. Also, this variation ranges from durations of 17.84 s to 21.01 s which is only an 17% variation. The mean of the durations is 18.36 s which is actually less than 3% from the lowest value so the distribution, as seen in Figure 7, is clearly skewed to local minima that are relatively close to the shortest trajectory of the set.

VI. Conclusion

Flight through urban environments presents challenges for motion planning due to the 3-D nature of the maneuvering required to traverse amongst close-proximity obstacles. Such planning can occur from the exploration of random dense trees into an environment using motion primitives that describe feasible maneuvers. In particular, the use of 3-D maneuvers as branches for 3-D trees is especially effective. It is shown that 2-primitive sequences are efficient at avoiding obstacles with a 3-primitive sequence traversing the final distance to the goal. An example of a representative scenario shows the feasibility of the approach to generate a motion plan for a micro air vehicle that needs to maneuver through a region of obstacles.

VII. Acknowledgements

This work was supported by ONR through grant N000014-07-1-0324 with Jason Stack and Mike Traweek as program monitors.

References

2. Faiz, N., Agrawal, S. and Murray, R., “Trajectory Planning of Differentially Flat Systems with Dynamics and


