Motion Primitive with Aerial Manipulator for Dynamic Tennis Swing Motion

Fan Shi Moju Zhao Tomoki Anzai Xiangyu Chen Kei Okada Masayuki Inaba

1. Introduction

In this paper, we implement a rapid real-time motion planning algorithm with a novel type of aerial manipulator, and dynamic tennis swing motion is achieved as an example to show the feasibility of the algorithm. When planning the feasible trajectory, dynamic constraints of the robot, and position, velocity and acceleration in start and end points are considered. Thousands of motion primitives are generated online and the least-cost trajectory to follow is selected by trajectory selector.

Recently, aerial robots raise increasing attentions and lots of fundamental researches appear in geometric control, 3D mapping, and real-time planning. With the help of these fundamental researches, aerial robots shows great potential to interact with environments and take more complex tasks such as multi-field drone, moving collaborate object. Among them, aerial manipulator is the most exciting and difficult one since it directly interact with the environments. Several researches of aerial manipulators are shown, some work attaches a high-DOFs robot arm to the downward base of the drone and more consideration in stability and workspace is needed[1], some are a simple 1-DOF gripper with limited functions[2]. Hence, our lab come up with a transformable drone (Hydrus Fig.1) as the aerial manipulator with dual arms, which have lots of potentials like moving in clustered environments and moving heavy objects[3][4].

In the following sections, we will design motion planning framework for Hydrus robot and use experiment result to verify the feasibility.

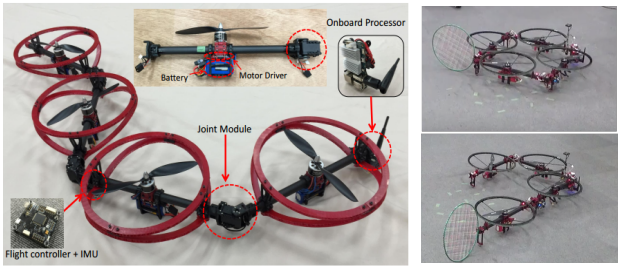

Fig.1 Left is the basic structure of 4-links Hydrus robot, which has 4 rotors and 3 motor-driven joint links. Right is Hydrus robot with tennis racket.

2. Dynamic Tennis Swing Motion

2.1 Model

Geometric control researches showed the differential flatness of quadrotor system[1], which means that the output of 4 rotors rotation speed could be calculated based on the given input $[s_x, s_y, s_z, \psi]$, here s_x, s_y, s_z, s_ψ is the function $s_x(t)$, $s_y(t)$, $s_z(t)$, ψ (t) representing the position of robot in x, y, z axis and yaw angle at time t. The dynamics of the translational motion and rotation motion can be described as follows:

$$M^{\{W\}}\ddot{\boldsymbol{r}}_{CoG} = \begin{bmatrix} 0 \\ 0 \\ -M^{\{w\}}g \end{bmatrix} + {}^{\{W\}}R \begin{bmatrix} 0 \\ 0 \\ \sum_{i=1}^{N} {}^{\{CoG\}}F_i \end{bmatrix} \tag{1}$$

$$\begin{aligned} {}^{\{CoG\}}I_{multilink} \begin{bmatrix} {}^{\{CoG\}}\dot{w}_x \\ {}^{\{CoG\}}\dot{w}_y \\ {}^{\{CoG\}}\dot{w}_z \end{bmatrix} &= \begin{bmatrix} \sum_{i=1}^{N} {}^{\{CoG\}}y_i {}^{\{CoG\}}F_i \\ -\sum_{i=1}^{N} {}^{\{CoG\}}x_i {}^{\{CoG\}}F_i \\ \sum_{i=1}^{N} {}^{\{CoG\}}T_i \end{bmatrix} \\ &- \begin{bmatrix} {}^{\{CoG\}}w_x \\ {}^{\{CoG\}}w_y \\ {}^{\{CoG\}}w_z \end{bmatrix} \times {}^{\{CoG\}}I_{multilink} \begin{bmatrix} {}^{\{CoG\}}w_x \\ {}^{\{CoG\}}w_y \\ {}^{\{CoG\}}w_z \end{bmatrix} \end{aligned} \tag{2}$$

2.2 Motion Primitive

In robotic trajectory planning researches, optimization and sampling based methods are two mainstreams. Optimization based methods is usually time consuming, so in this paper we develop a rapid motion primitive algorithm based on sampling methods.

We develop a motion primitive method based on differential flatness quality, and 6-order polynomial functions are calculated in every axis and keep yaw angle to be fixed in the whole trajectory. Then jerk will be evaluated as the smoothness of the trajectory[5].

The motion primitive in each axis follows:

$$\boldsymbol{f}(t) = \begin{bmatrix} \boldsymbol{s}(t) \\ \boldsymbol{v}(t) \\ \boldsymbol{a}(t) \end{bmatrix} \tag{3}$$

$$\boldsymbol{f}(t) = \begin{bmatrix} \boldsymbol{a} & \boldsymbol{b} & \boldsymbol{c} & \boldsymbol{d} & \boldsymbol{e} & \boldsymbol{f} \\ 0 & 5\boldsymbol{a} & 4\boldsymbol{b} & \boldsymbol{c} & 2\boldsymbol{d} & \boldsymbol{e} \\ 0 & 0 & 20\boldsymbol{a} & 12\boldsymbol{b} & 6\boldsymbol{c} & 2\boldsymbol{d} \end{bmatrix} \begin{bmatrix} 1 \\ t \\ t^2 \\ t^3 \\ t^4 \\ t^5 \end{bmatrix} \quad (4)$$

For any random start point, we could acquire its position, velocity and acceleration from robot inertial data. The end point would be the hitting point, and its position, velocity and acceleration could be sampled in the work space. Considering a simple situation, we assume tennis ball is fixed, so only velocity and acceleration are needed to be sampled. For traverse time, we use traverse trajectory distance and average traverse speed to get the traverse time.

To get the faster swing speed, motion primitive for swing joint is also generated(Fig.2). The racket attached joint will backswing in advance, and then swing to guarantee racket will have velocity when being in collision with tennis ball. The backswing and swing time is decided by the selected trajectory of the robot.

2.3 Trajectory Selection

Considering the dynamic feasibility of the robot, the maximum velocity and acceleration have to be guaranteed. So the selector is as:

$$\begin{aligned} & min \int_0^T |s_x^{(3)}(t) + s_y^{(3)}(t) + s_z^{(3)}(t)|^2 dt \quad (5) \\ s.t. \;\; & V_{i_{min}} < s_i^{(1)}(t) < V_{i_{max}}, i = x, y, z \\ & a_{i_{min}} < s_i^{(2)}(t) < a_{i_{max}}, i = x, y, z \\ & 0 < s_z(t) \end{aligned}$$

Objective function is showed as Eq.(5), the calculus of the jerk ṡ absolute value from start time to end time is calculated. The following constraints guarantee the dynamic constraints of robot and robot not crashing into the ground.

If there are other obstacles except for the ground in the environment, collision-free constraints should be considered as:

$$C_{obstacle} \bigcap (s_x(t), s_y(t), s_z(t)) = \emptyset \quad (6)$$

3. Simulation Result

We implement this method into Gazebo simulation and nearly 5, 000 motion primitives could be generated in 100ms, and feasible trajectory could be easily found inside these candidates Fig.2. Then we build position and velocity control loop to guarantee the robot could follow the feasible trajectory.

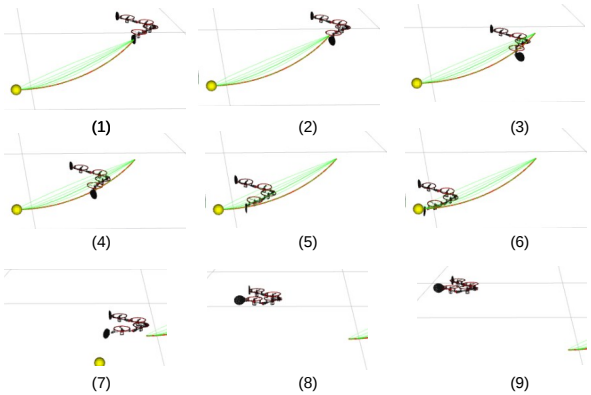

Fig.2 (1)-(6) is the swing motion based on motion primitive, green lines are primitive candidates, red line is the selected candidate, yellow ball is the hitting target. (1) is the start point, (2) begins to do backswing, (3) finishes backswing, (4) starts swing motion, (5) gets close to target and continue swing motion, (6) hits the target ball. (7)-(9) is the attitude control after hitting motion, robot recovers to be still and specific height.

4. Conclusion

This paper showed a rapid motion planning algorithm for Hydrus robot, and implement tennis swing task as the example to show the feasibility of the method. Experiment is implemented in simulation and the whole system is evaluated.

For future work, we are expecting to implement the method into real physical Hydrus robot to evaluate the whole system. And the use of GPU to accelerate motion primitive calculation is also considered. Perception of the tennis ball to generate catch-and-swing motion will be implemented as the future work.

Reference

[1] A. E. Jimenez-Cano, J. Martin, G. Heredia, A. Ollero, and R. Cano. Control of an aerial robot with multi-link arm for assembly tasks. In *2013 IEEE International Conference on Robotics and Automation*, pp. 4916–4921, May 2013.

[2] P. M. Dames, M. Schwager, D. Rus, and V. Kumar. Active magnetic anomaly detection using multiple micro aerial vehicles. *IEEE Robotics and Automation Letters*, Vol. 1, No. 1, pp. 153–160, Jan 2016.

[3] Moju Zhao, Koji Kawasaki, Kei Okada, and Masayuki Inaba. Transformable multirotor with two-dimensional multilinks: modeling, control, and motion planning for aerial transformation. *Advanced Robotics*, Vol. 30, No. 13, pp. 825–845, 2016.

[4] Moju Zhao, Koji Kawasaki, Xiangyu Chen, Yohei Kakiuchi, Kei Okada, and Masayuki Inaba. Transformable multirotor with two-dimensional multilinks: Modeling, control, and whole-body aerial manipulation. In *International Symposium on Experimental Robotics*, pp. 515–524. Springer, 2016.

[5] Mark W Mueller, Markus Hehn, and Raffaello D'Andrea. A computationally efficient algorithm for state-to-state quadrocopter trajectory generation and feasibility verification. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 3480–3486. IEEE, 2013.