Available online at www.sciencedirect.com
ScienceDirect
IFAC-PapersOnLine 48-9 (2015) 257–262

Model Predictive Control with Ellipsoid Obstacle Constraints for Spacecraft Rendezvous

Christopher Jewison * **R. Scott Erwin** ** **Alvar Saenz-Otero** ***

* *Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: jewisonc@mit.edu)*
** *Air Force Research Laboratory, Space Vehicles Directorate, Kirtland Air Force Base, NM 87117 USA*
*** *Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: alvarso@mit.edu)*

Abstract: The problem of spacecraft rendezvous with obstacle avoidance constraints is explored. A Model Predictive Control (MPC) approach is used to compute an optimal control strategy for a chaser attempting to rendezvous with a target spacecraft in Earth orbit. Given obstructions to the baseline optimal trajectory, such as orbital debris or other spacecraft, MPC attempts to update the trajectory in real time such that it evades these obstacles. In this work, obstacles are approximated or bounded by ellipsoids to both enable straightforward constraint evaluation and better represent statistical knowledge of the obstacle's position. A nonlinear optimization method, Sequential Quadratic Programming, is able to solve this quadratic optimal control problem with nonlinear obstacle avoidance constraints. Specifically, the cases of multiple and moving obstacles are handled well with this approach due to the flexibility of the nonlinear constraint formulation. Implementation of this algorithm and results from a MATLAB-based simulation are discussed. This ellipsoid constraint approach is compared to a previous method involving a convex, rotating hyperplane constraint. The nonlinear programming approach presented is more computationally expensive than previous methods seen in the literature, but shows markedly improved results in a few key areas.

Keywords: Obstacle avoidance; trajectory planning; optimal control; nonlinear programming; quadratic programming; spacecraft autonomy.

1. INTRODUCTION

Traditionally, spacecraft rendezvous trajectory planning is performed open loop in the mission planning phase and a closed-loop controller is used to track that pre-planned trajectory. However, there are over 21,000 pieces of debris larger than 10 cm currently being tracked in orbit around the Earth.[1] Given the possibility of potentially mission-ending collisions, there is a need for the planning and constant updating of safe trajectories for high-valued space assets, such as manned vehicles. In addition, spacecraft lifetimes are highly limited by fuel consumption; thus, performing these maneuvers propellant-optimally would extend spacecraft missions.

There has been significant work performed in the field; however, there are several areas open for improvement, and the field still remains active. Several methods have been employed in literature for the past decade. Some of these methods have only been implemented for open-loop trajectory planning and do not have convergence guarantees or fast-enough computation times to be implemented in a real-time system. Genetic algorithm techniques have been used for open-loop trajectories, however can require more than a day of computation as seen in Luo et al. (2007a). Other promising methods include Second Order Cone Programming, Lu and Liu (2013); Rapidly-exploring Random Trees, Garcia and How (2005); Simulated Annealing, Luo et al. (2007b); and Sequential Quadratic Programming (SQP), Luo et al. (2007b). In the literature, these listed methods are not implemented in real-time systems and, to this point, have only been used for mission planning purposes.

Real-time requirements have, however, been met using convex, linear optimization techniques. Petersen et al. (2014) worked on a method of trajectory planning using an optimal control method called Model Predictive Control (MPC), which is tailored to real-time systems. Inside this control architecture, an obstacle-avoidance technique linearizes the nonlinear, non-convex obstacle constraint into a rotating hyperplane that is convex and has associated convergence guarantees. Also shown in Di Cairano et al. (2012), this rotating hyperplane constraint permits easy implementation with linear or quadratic optimizers. Additionally, Mixed Integer Linear Programming, described by Richards et al. (2002), allows real-time implementation by using a set of binary weights to switch different convex constraints on and off along the trajectory. Due to these linearizations, the methods yield conservatively planned

[1] http://orbitaldebris.jsc.nasa.gov/faqs.html

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2015.08.093

trajectories, yet are computationally fast. The main aim of this paper is to implement nonlinear, ellipsoidal, obstacle avoidance constraints in a real-time, MPC framework.

This paper is organized as follows. Sections 2 and 3 describe the relative motion dynamics, the trajectory optimization problem to be solved, and the MPC formulation, relying heavily on the work of Petersen et al. (2014). In Section 4, we present the main contribution of the current paper by reformulating the constraints of the problem from the linear, convex approximation used by Petersen et al. (2014) into a nonlinear, nonconvex constraint that is solvable within the SQP format. In Section 5, we discuss the results from a MATLAB-based simulation in comparison to the work performed by Petersen et al. (2014).

2. SPACECRAFT RELATIVE MOTION

The Hill-Clohessy-Wiltshire (HCW) frame is used for the two-spacecraft rendezvous problem. Orbital mechanics govern the relative motion of spacecraft according to the linearized equations from Clohessy and Wiltshire (1960):

$$\begin{aligned} \ddot{x} - 2n\dot{y} - 3n^2 x &= \frac{F_x}{m}, \\ \ddot{y} + 2n\dot{x} &= \frac{F_y}{m}, \\ \ddot{z} + n^2 z &= \frac{F_z}{m}, \end{aligned} \tag{1}$$

where $n = \sqrt{\frac{\mu}{{r_o}^3}}$ is the mean motion of the target satellite, with μ as the standard gravitational parameter and r_o as the orbital radius of the target spacecraft in a circular orbit; x is the radial position, y is the in-track position, and z is the cross-track position of the chaser satellite with respect to the target satellite at the origin; m is the mass of the chaser spacecraft; and F_x, F_y, and F_z are the actuated forces of the chaser.

The HCW equations in Equation 1 are converted into the state-space form shown in Equation 2, as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3n^2 & 0 & 0 & 0 & 2n & 0 \\ 0 & 0 & 0 & -2n & 0 & 0 \\ 0 & 0 & -n^2 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{1}{m} & 0 & 0 \\ 0 & \frac{1}{m} & 0 \\ 0 & 0 & \frac{1}{m} \end{bmatrix} \begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}. \tag{2}$$

Furthermore, the state-space form in Equation 2 can be discretized for a specific sampling time both in a continuous and impulsively controlled spacecraft. The discretized form of the system is

$$\mathbf{x}_{k+1} = \mathbf{A}_d \mathbf{x}_k + \mathbf{B}_d \mathbf{u}_k, \tag{3}$$

where $\mathbf{A}_d$ and $\mathbf{B}_d$ are the discretized matrices defined in Petersen et al. (2014), $\mathbf{x}_{k+1}$ is the propagated state after the discretized sampling period, $\mathbf{x}_k$ is the current state and $\mathbf{u}_k$ is the control input defined as either ΔV or axial forces. Note that the $\mathbf{B}_d$ matrix changes depending on choice of continuous (axial forces) or impulsive (ΔV) control.

3. MODEL PREDICTIVE CONTROL

MPC, an optimal control technique, uses discretized dynamics to step through a prediction horizon. Each step in the horizon is the length of the discretized sampling period. Thus, the prediction horizon consists of a set of control inputs (thruster firings) and states at each of the specified times in the future. At each execution time, an optimal control problem is solved to find the new prescribed control for the entire horizon. The spacecraft then implements only the first control step. At the next step, the optimal control problem is solved again, and the first step in this new solution is implemented. This process repeats until the spacecraft reaches termination conditions such as a distance of 50 m or completed rendezvous. MPC allows for the recomputation of a trajectory given unexpected obstacles move into the initial trajectory. In this manner, as long as the spacecraft is aware of a new obstacle, the MPC method is robust to a variety of unexpected constraints popping up at a moments notice.

For this work, the MPC objective function to be optimized at each step is detailed in Petersen et al. (2014) and Brand et al. (2011). In this formulation, known as Parallel Quadratic Programming (PQP), the states, $\mathbf{x}_k$, for each time step in the horizon are stacked into a long column vector, $\bar{\mathbf{X}}$, and the control inputs, $\mathbf{u}_k$, into $\bar{\mathbf{U}}$. The propagation of dynamics can be represented as linear constraints in the HCW form shown in Equation 2. Additional linear constraints on the maximum axial thruster force or ΔV can be concatenated to the dynamics constraints. The optimal control problem, without obstacle avoidance constraints, can thus be posed in the form of

$$\begin{aligned} \min_{\bar{\mathbf{U}}} \ & \frac{1}{2}\bar{\mathbf{U}}^T \mathbf{S} \bar{\mathbf{U}} + \mathbf{H}^T \bar{\mathbf{U}}, \\ & \text{subject to} \\ & \mathbf{V}\bar{\mathbf{U}} \le \mathbf{W}, \end{aligned} \tag{4}$$

where $\mathbf{S}$, $\mathbf{H}$, $\mathbf{V}$, and $\mathbf{W}$ are are left undefined here simply to show the structure of the optimal control problem, but are formulated for PQP as discussed in Brand et al. (2011).

Equation 4 sets up the optimal control problem without the obstacle avoidance constraints added, which, up to this point, is identical to that described in Petersen et al. (2014). Implementing this linearly constrained, quadratic optimal control problem in Equation 4 with SQP in MATLAB yielded the same trajectory as PQP output in Petersen et al. (2014), although the SQP had slower computation time by a factor of 15 (about 135 vs. 9 ms). PQP is able to solve a quadratic problem with linear constraints very quickly, while SQP is tailored toward nonlinear optimization and thus has some extra overhead. More results will be discussed once the obstacle constraints have been introduced in the following section.

4. OBSTACLE AVOIDANCE

As discussed in the introduction, avoiding obstacles in the chaser's trajectory is very important to the success of a mission. There are several established methods for orbital debris avoidance in the literature. One of the more prominent techniques and the one implemented in Petersen et al. (2014) uses a hyperplane constraint that slowly rotates