Autonomous Spacecraft Rendezvous and Docking Literature Review

Dhruv Jain
School of Aeronautics and Astronautics
Purdue University
West Lafayette, U.S.A
jain220@purdue.edu

Dr. James Goppert
School of Aeronautics and Astronautics
Purdue University
West Lafayette, U.S.A
jgoppert@purdue.edu

***Abstract*—This paper presents a review of the previous efforts to produce a control scheme for Autonomous Spacecraft Rendezvous and Docking of a spacecraft with another spacecraft. An efficient way to represent the dynamics of the two spacecrafts is reviewed along with a few methods to describe the dynamics of the two. In addition, different state estimation algorithms and path planning algorithms are compared. Also, a brief overview of a few methods which can be used to simulate the environment to check the control scheme is discussed.**

***Index Terms*—Autonomous Spacecraft Rendezvous, Docking, Lie algebra, Gazebo**

I. INTRODUCTION

In the 21st Century, access to space has become easier and there is a steady increase in the number of satellites and spacecraft launched every year. However, one of the key capabilities lacking in the increasing number of parties involved is an efficient Spacecraft Docking system. A Spacecraft Docking system can help to increase the lifetime of a satellite by refueling or on-orbit servicing, large-scale structure assembly, supply to ISS and future space stations, and reducing space debris.[1]

The dynamics of the target spacecraft and chaser spacecraft are very difficult to describe and the control scheme needs to adapt to continuous disturbances due to various varying orbital perturbations as well as the change in mass of the spacecraft due to the frequent use of thrusters. There is a need to represent the dynamics of the spacecraft in a concise manner without loss of information for ease of understanding and performing the necessary transformations. Alongside this, a mathematical representation is required that is not susceptible to singularities. This paper will review the necessary mathematical representation, different path planning algorithms, state estimation algorithms, and control schemes proposed to tackle the problem. Thereafter, a few methods to simulate the control scheme are discussed.

This paper is focused on spacecraft docking, however, the setup can be expanded and the control system can be slightly modified for spacecraft rendezvous.

II. SETUP AND ASSUMPTIONS

A spacecraft is considered to be in docking if it comes as close as 100m to the target [2]. The process of docking is assumed to mean that the chaser and target will mechanically connect when the chaser is propelled at a finite linear velocity [11]. It is important that in the process of docking, the chaser's attitude is described in a system that is not susceptible to singularities so that the control estimation algorithm runs smoothly. The spacecraft's Docking control system has to be very sensitive as it is essential that the alignment between the two spacecraft is perfect.

In the setup, the chaser has the capability to control the 6 degrees of freedom (DOF). It has a main motor at the bottom and 8 thrusters. 4 out of the 8 thrusters are equally spaced and placed toward the bottom, and the other 4 are equally spaced and placed toward the top of the spacecraft. The 8 thrusters provide the capability to be used in different combinations to control the 6 DOF. The target is considered to have a fixed rate change of attitude and velocity, and it is assumed to be a passive target. A passive target is one that cannot control its 6 DOF and is not tumbling.[11][12]

The spacecrafts are located at 500 km altitude so that the control system can be modified to be used for docking a spacecraft to ISS if needed. The spacecrafts are assumed to have a very small relative distance compared to their geocentric distance, so all the orbital perturbations are ignored. It is also assumed that the mass change of the chaser during the process is negligible, so the center of mass is fixed. Thus, the relative motion of the two spacecrafts can be defined using the Clohessy-Wiltshire equation. Since the Clohessy-Wiltshire equation is linear so the relative motion of the spacecraft is linear. However, the attitude dynamics of the spacecraft is governed by a non-linear set of equation. Furthermore, the trajectory of the chaser is known before it actually starts following it so the system can be interpreted as time-invariant as the trajectory is not changing with time.

It is even assumed that the spacecraft's mechanism does not fail, so the FDIR algorithm is not used.

III. MATHEMATICAL REPRESENTATION OF SPACECRAFT DYNAMICS

Spacecrafts have 6 degrees of freedom (3 translational and 3 rotations). These vectors change with time due to various perturbations and forces exerted by the spacecraft's thrusters. To effectively describe the dynamics, 5 important frames are considered:

- ECI (Earth-Centered Inertial) Frame
- Chaser Spacecraft Frame
- Target Spacecraft Frame
- Chaser Spacecraft's Orbital Plane
- Target Spacecraft's Orbital Plane

Due to multiple frames and vectors, there is a dire need to concisely represent all the information.

The Lie group SE(3) can succinctly represent the dynamics of a spacecraft. It is a 4x4 matrix that contains the rotation matrix describing the frame under consideration with respect to a frame common to the two spacecraft, and its position vector [1]. The SE(3) has two subgroups SO(3), which is the rotation matrix, and T(3), which contains the position vector. Lie Algebra can be used on elements of a complex group such as the semi-direct product of SE(3) with T(3) or some other form to easily change the coordinate frame as well as describe the attitude of the spacecraft at any point of time t. However, a drawback of using the Lie group is that all the components should be group affine, and setting up the group can sometimes add complexity for instance there may exist a left variant as well as a right variant component, which makes the calculation of X^{-1} $\acute{X}$ difficult.

IV. SPACECRAFT ATTITUDE DESCRIPTION

There is a need to describe the spacecraft attitude in a system that does not have singularities or has a method to overcome them and has fewer parameters for faster integration.

The 4 popular systems to describe the Spacecraft Attitude and their shortcomings are:

- Directional Cosine Matrix (DCM): The system has 9 parameters, so it is difficult to set up and it is computationally very expensive to integrate. However, it does not have any singularity.
- Euler Angles: The system has 3 parameters but it has a singularity at 90^{o}. Thus, it cannot be used for describing the spacecraft's attitude.
- Quaternions: The system has 4 parameters and has no singularity.
- Modified Rodriquez Parameters(MRP): The system has 3 parameters but it has a singularity at 180 degrees. However, the problem can be overcome by using their shadow set, so another parameter is used to keep track if the MRP is in the shadow set or not. Thus, MRP is considered to have 4 parameters and no singularity.

Since, Quaternions and MRP are not susceptible to singularity and have 4 parameters, which is far less than the number of parameters of DCM, it is computationally viable to choose either of the two systems. Since there is no difference in ease of use between Quaternions and MRP, Quaternions are chosen as the viable option to describe a spacecraft's orientation.

V. PATH PLANNING ALGORITHMS

Given the initial state of the two spacecrafts i.e. position, velocity, acceleration, swap, and attitude, there is a need to find a viable trajectory that the chaser can follow to reach the target. In a general sense, a path or trajectory is a collision-free set of geometric points that connects the two points and the locus of points may be constrained to time, energy, or some other parameter. [13]

The two important considerations for an optimal trajectory are a low amount of fuel to be used and the two should have the same line of sight. At the same time, it has to be computationally viable for on-board processing.

Underwater Vehicles act in a manner very similar to spacecraft, so their Path Planning Algorithms were also explored. But due to the different setups very different forces act on the body which drastically changes the problem and thus the purpose of the algorithm. So, the focus was narrowed only to the literature on spacecraft applications.[14]

There are 3 path-planning algorithms popularly cited for spacecraft rendezvous and docking. The 3 are:[12]

- Mixed Linear Integer Programming(MLIP): It is an efficient method of solving constrained linear optimization problems. However, it is computationally very expensive, so it is not a viable option for a real-time system. MLIP can be used for spacecraft trajectory planning if the calculations are done offline and a very ideal case is considered.
- Model Predictive Control(MPC): It is built to serve discrete linear time-invariant systems with constraints on states and inputs. A major benefit of this algorithm is that the computation can be reduced to a simple function evaluation, so it can be processed onboard in real-time. However, it is very complex to set up and obtain the simplified function. In addition, its performance exponentially decreases with an increase in dimensionality, so a simpler algorithm can be used in place of it.[21]
- Glideslope algorithm: It is the most widely used and proven path planning algorithm for spacecraft trajectory planning for relatively short distances. It was used for Apollo and Shuttle as it can be used in real-time and

it can be used for evasive maneuver planning. It is comparatively simple, robust, and easy to implement, but it is not always optimal. Another drawback of the real-life application is that it does not have a collision avoidance feature. However, due to its characteristic of reducing induced velocity at each burn, it can help reduce the effects of plume impingement.[24]

A relatively new idea that has not been widely implemented is Multi-Manuever Clohessy-Wiltshire Targeting. This method involves manipulating CW state matrix to form a linear system. It can effectively be used to find discrete maneuvers required for a chaser to reach the target. This idea is fairly complex to understand as a whole and implement but can be exploited in the future due to its ease of finding individual maneuver ΔV.[15]

The above-discussed methods are direct methods, so an indirect method can even be used to plan an optimal trajectory.

A cost function for given control input and dynamic constraint can be formulated, and Euler-Lagrange Equations can be used to come up with a variational trajectory optimized for fuel use. The paper written by C. Henshaw and R. Sanner explains the mathematics of finding the cost function and solving it for different constraints.[16]

From the 3 Direct Methods and 1 Indirect Method discusses above, the intelligent choice to plan a trajectory that is not complex to compute and implement is either to use Glideslope Algorithm or to use the indirect method. Another advantage of using either of the two shortlisted options is that both have been proven to work in the real-world.

VI. STATE ESTIMATION ALGORITHMS

After the crucial step of knowing which trajectory the chaser should follow, it is important to estimate its state in order for the control system to be effective to keep the spacecraft on its required trajectory.

There are two main algorithms considered: Kalman Filter and Particle Filter.

Kalman Filter is an optimal algorithm for a linear system with Gaussian noise, so it produces poor results for the non-linear system. Thus it cannot be used for attitude estimation, so an improvement of the Kalman Filter known as the Extended Kalman Filter(EKF) can be used as it considers the first-order term of a non-linear system. However, EKF provides poor accuracy as it just considers the first order term, so a better version of EKF exists known as Unscented Kalman Filter(UKF). UKF considers the first-order and second-order terms which makes it more accurate. It is the most widely used state estimation algorithm for spacecraft due to its fairly high accuracy and low computation cost. UKF even leads to faster convergence from inaccurate initial conditions due to its setup.[17-20]

Particle Filter(PF) is another popular state estimation algorithm, widely used for non-linear systems. It is a more general form of UKF and can be used to effectively deal with highly non-linear systems. ”PF is estimated through weighted particles(random particles) that are generated with the pseudo-random generator.”[12] A major drawback of Particle Filter is that it is heavily dependent on a number of particles, and so is their computational cost.[18]

Since the system under consideration is not highly non-linear and the on-board processing is limited, so a computationally efficient algorithm that can deal with the non-linear systems without trading off accuracy is required. The only algorithm that fits the description amongst the algorithms discussed is the Unscented Kalman Filter. Thus, UKF will be used for state estimation.

There exists another state estimation algorithm that is widely used called Moving Horizon Estimation but due to its complexity to understand and implement it was not discussed.

VII. POPULAR CONTROL SCHEMES

A spacecraft often encounters noise due to multiple approximations in the process of state estimation and disturbance due to multiple small forces which were neglected. So, a spacecraft diverges from the required trajectory and there is a need for a control system that can guide the spacecraft back to the required trajectory.

The data about the attitude of the spacecraft can be obtained through a gyroscope, magnetometer, star tracker, and accelerometer. However, these methods are imprecise as docking requires very high precision, so often the chaser has a LIDAR or CCD camera, to obtain an image to be as used as input to obtain the relative distance and orientation of the two spacecraft [4-5].

The fundamental equation which describes the relative motion between two spacecraft that are close to each other is given by Clohessy-Wiltshire Equation.[3]

1) $\ddot{x} = 3n^2x + 2n\dot{y}$
2) $\ddot{y} = -2n\dot{x}$
3) $\ddot{z} = -n^2z$
4) $n = \sqrt{\mu/a^3}$

Where a is the semi-major axis.

The set of equations that describes the attitude of a spacecraft is given by the derivative of attitude represented by quaternions.

The last step to building an autonomous spacecraft docking system is to use a control scheme that can keep the spacecraft on its reference trajectory and as noted earlier the dynamics of the spacecraft is linear but the attitude is non-linear, nevertheless, the system is time-invariant.

Since the early space race, NASA used manual control for docking, and was not until the late 1990s and early 2000s that NASA shifted to the automatic docking control system. Whereas Roscosmos tested an autonomous system through a spacecraft called Igla in 1967, they continued to improve upon the control system through Salyut missions and they played a key role in the construction of ISS. Interestingly, the docking control system used by the space organization is not public knowledge, so one can only speculate which control system they used. However, extensive knowledge of multiple control systems exists for a different set of assumptions, which engineers have tested through different simulations.[12]

There are 4 popular control schemes that people have simulated to prove that they work for spacecraft docking. Those 5 are:[12]

1) PID control: The basic form of PID control only works when the non-linearities and discontinuities are small, so that the system can be approximated as a linear and continuous system. However, PID controllers can be made robust and non-linear, and the non-conventional form of PID controller is widely used for Spacecraft attitude control.[22][23]
2) Fuzzy Control: It is to use a Genetic Algorithm to perform optimization of the fuzzy controller by finding the best fuzzy set of the member function to optimize the docking time and the fuel consumption [2]. The fuzzy control is meant to take the state of the spacecraft as input and calculate the required torque to achieve the required trajectory. Alongside, the General Algorithm tool can be defined to run Reproduction, Crossover, and Mutation to optimize the membership function of fuzzy control. This control scheme is much more computationally expensive and difficult to set up than PID control but it efficiently helps the spacecraft to correct the various noises and disturbances.
3) Phase plane control: It is another widely used controller in the space industry and it computes state errors to generate pre-determined thrust pulses. Another feature of this controller is that each DOF is independently controlled, so it is much easier to set up and understand the results.
4) LQR: It is a very popular and optimal control scheme. Through different simulations, it is shown to produce more fuel-efficient correction and if it is used with a pulse-width modulator then it can produce more accurate results. Its setup is very similar to PID control, so LQR and PID control have marginal differences in feasibility.

From the discussion above, the three best controllers are PID control, Phase Plane control, and LQR. Further analysis shows that the Phase Plane Control is optimal against temporary errors and it leads to results that are more fuel efficient than PID control. So, between Phase Plane control and LQR, since Phase Plane control's results are easier to interpret and to set up, thus Phase Plane control seems to be the best control scheme for the setup.

VIII. SIMULATION

It is difficult to replicate the real scenario to test the control scheme. However, a simplified scenario can be mimicked where two scaled vehicles can be put on a low-friction table and the control scheme can be tested for 3 degrees of freedom(2 Translation and 1 Rotation) [10]. Since this is not a sufficient solution to test the control scheme so a program can be made by setting a scenario to see if the position of the chaser converges with that of the target. But this does not provide a visualization of the process. Thus a physics engine such as Gazebo can be used to test the control scheme and get a visualization of the process.

Gazebo is a widely used software tool for robotic simulation and only one project related to spacecraft has been simulated in Gazebo. A robot called Astrobee was developed by NASA to be used in ISS and its dynamics were simulated in Gazebo. Thus, the control scheme which will be developed will be simulated in Gazebo by making plugs-in Casadi. A random gaussian number generator of normalized magnitude will be used to introduce errors into the system to check the performance of the controller.

IX. CONCLUSION

Spacecraft Rendezvous and Docking are crucial features that every upcoming spacecraft should have to have a longer life and have the capability to be used for a different purpose in the future if required. However, this paper and further efforts are focused on spacecraft docking but the same setup can later be expanded to be used for spacecraft rendezvous.

An efficient method to represent the dynamics of a spacecraft is by using the SE(3) Lie group, however, it can make the calculation more complex due to its limitation. As discussed above Quaternions will be used for describing the spacecraft's attitude. An important algorithm required for the architecture is the path planning algorithm, there are two viable options: the Glideslope algorithm and the Euler-Lagrangian equations. The next key piece is a feasible state estimation algorithm and as discussed above Unscented Kalman Filter will be used. After, a careful comparison of different control schemes it is concluded that Phase Plance control will serve as the best controller for the setup. The control scheme can then be tested by simulating the environment in Gazebo by using plug-ins made using Casadi in Python.

One of the key things that need to be worked upon is to find a viable Thruster Management Algorithm. It is an algorithm that can convert the necessary forces and momentum found from previous calculations into forces that individual thrusters should exert so that their combination can lead to the required output.

REFERENCES

[1] D. Lee and G. Vukovich, Robust adaptive terminal sliding mode control on SE(3) for autonomous spacecraft rendezvous and docking, Nonlinear Dyn., vol. 83, no. 4, pp. 22632279, 2016.

[2] R. Bosch, D.- Darmstadt, and J. M. Giron-sierra, GENETIC ALGORITHMS FOR FUZZY CONTROL OF.

[3] R. Kristiansen, E. I. Grtli, P. J. Nicklasson, and J. T. Gravdahl, A model of relative position and attitude in a leader-follower spacecraft formation, Model. Identif. Control, 2007

[4] J. R. Wertz and R. Bell, Autonomous rendezvous and docking technologies: status and prospects, Sp. Syst. Technol. Oper., vol. 5088, no. August 2003, pp. 2030, 2003.

[5] P. Singla, K. Subbarao, and J. L. Junkins, Adaptive output feedback control for spacecraft rendezvous and docking under measurement uncertainty, J. Guid. Control. Dyn., vol. 29, no. 4, pp. 892902, 2006.

[6] E. Capello, E. Punta, F. Dabbene, G. Guglieri, and R. Tempo, Sliding-mode control strategies for rendezvous and docking maneuvers, J. Guid. Control. Dyn., vol. 40, no. 6, pp. 14811488, 2017.

[7] C. Tournes and Y. Shtessel, Automatic docking using second order sliding mode control, Proc. Am. Control Conf., pp. 38553860, 2007.

[8] N. K. Philip and M. R. Ananthasayanam, Relative position and attitude estimation and control schemes for the final phase of an autonomous docking mission of spacecraft, Acta Astronaut., vol. 52, no. 7, pp. 511522, 2003

[9] S. Diemer and S. Bonnabel, An invariant Linear Quadratic Gaussian controller for a simplified car, Proc. - IEEE Int. Conf. Robot. Autom., vol. 2015-June, no. June, pp. 448453, 2015.

[10] G. Guglieri, F. Maroglio, P. Pellegrino, and L. Torre, Design and development of guidance navigation and control algorithms for spacecraft rendezvous and docking experimentation, Acta Astronaut., 2014.

[11] Polites, Michael E. Technology of Automated Rendezvous and Capture in Space. Journal of Spacecraft and Rockets, vol. 36, no. 2, 1999, pp. 280291., doi:10.2514/2.3443.

[12] Simon,, Nolet,. (2007). Development of a guidance, navigation and control architecture and validation process enabling autonomous docking to a tumbling satellite.

[13] Gasparetto, Alessandro, et al. Path Planning and Trajectory Planning Algorithms: A General Overview. Motion and Operation Planning of Robotic Systems Mechanisms and Machine Science, 2015, pp. 327

[14] C. Petres, Y. Pailhas, P. Patron, Y. Petillot, J. Evans and D. Lane, ”Path Planning for Autonomous Underwater Vehicles,” in IEEE Transactions on Robotics, vol. 23, no. 2, pp. 331-341, April 2007.

[15] Richards, Louise M. MULTI-MANEUVER CLOHESSY-WILTSHIRE TARGETING. Reference Reviews, vol. 22, no. 8, 2008, pp. 4041., doi:10.1108/09504120810914619.

[16] Henshaw, Carl Glen, and Robert M. Sanner. Variational Technique for Spacecraft Trajectory Planning. Journal of Aerospace Engineering, vol. 23, no. 3, 2010, pp. 147156., doi:10.1061/(asce)as.1943-5525.0000019.

[17] Smith, Laurence. Sequential Monte Carlo Particle Filtering for State Estimation. doi:10.22215/etd/2006-08232.

[18] Crassidis, John L., and F. Landis Markley. Unscented Filtering for Spacecraft Attitude Estimation. Journal of Guidance, Control, and Dynamics, vol. 26, no. 4, 2003, pp. 536542., doi:10.2514/2.5102.

[19] Vandyke, Matthew Schwartz, Jana Hall, Christopher. (2004). Unscented Kalman Filtering for spacecraft attitude state and parameter estimation. Advances in the Astronautical Sciences. 119.

[20] Kim, Son-Goo, et al. Kalman Filtering for Relative Spacecraft Attitude and Position Estimation. Journal of Guidance, Control, and Dynamics, vol. 30, no. 1, 2007, pp. 133143., doi:10.2514/1.22377.

[21] M. Nolte, M. Rose, T. Stolte and M. Maurer, ”Model predictive control based trajectory generation for autonomous vehicles An architectural approach,” 2017 IEEE Intelligent Vehicles Symposium (IV), Los Angeles, CA, 2017, pp. 798-805. doi: 10.1109/IVS.2017.7995814

[22] Li, John H. Design, Analysis, and Simulation of Attitude Controllers for the Micro Nimbus Mission.

[23] Kumar, Supanna S., et al. Robust PID Controller Design for Rigid Uncertain Spacecraft Using Kharitonov Theorem and Vectored Particle Swarm Optimization. International Journal of Engineering Technology, vol. 7, no. 2.21, 2018, p. 9., doi:10.14419/ijet.v7i2.21.11825.

[24] Hablani, Hari B., et al. Guidance and Relative Navigation for Autonomous Rendezvous in a Circular Orbit. Journal of Guidance, Control, and Dynamics, vol. 25, no. 3, 2002, pp. 553562., doi:10.2514/2.4916.