Acta Astronautica 68 (2011) 1260–1275

Contents lists available at ScienceDirect

Acta Astronautica

journal homepage: www.elsevier.com/locate/actaastro

Development and experimentation of LQR/APF guidance and control for autonomous proximity maneuvers of multiple spacecraft

R. Bevilacqua [a,*], T. Lehmann [b], M. Romano [b]

[a] Department of Mechanical, Aerospace, and Nuclear Engineering, Rensselaer Polytechnic Institute, Jonsson Engineering Center, 110 8th Street, Troy, NY 12180-3590, United States
[b] Mechanical and Aerospace Engineering Department, code MAE/MR, 700 Dyer Rd., Naval Postgraduate School, Monterey, CA 93943, United States

ARTICLE INFO

Article history:
Received 3 July 2009
Received in revised form
31 July 2010
Accepted 5 August 2010
Available online 15 September 2010

Keywords:
Multiple spacecraft assembly
Autonomous control
Linear quadratic regulator
Artificial potential function
Wall-following method
Real-time control
Laboratory experimentation
On-the-ground spacecraft simulation
Autonomous on orbit operations
Formation flight
Spacecraft servicing

ABSTRACT

This work introduces a novel control algorithm for close proximity multiple spacecraft autonomous maneuvers, based on hybrid linear quadratic regulator/artificial potential function (LQR/APF), for applications including autonomous docking, on-orbit assembly and spacecraft servicing. Both theoretical developments and experimental validation of the proposed approach are presented. Fuel consumption is sub-optimized in real-time through re-computation of the LQR at each sample time, while performing collision avoidance through the APF and a high level decisional logic. The underlying LQR/APF controller is integrated with a customized wall-following technique and a decisional logic, overcoming problems such as local minima. The algorithm is experimentally tested on a four spacecraft simulators test bed at the Spacecraft Robotics Laboratory of the Naval Postgraduate School. The metrics to evaluate the control algorithm are: autonomy of the system in making decisions, successful completion of the maneuver, required time, and propellant consumption.

1. Introduction

The ability of multiple spacecraft systems to autonomously track, rendezvous, inspect, and dock has many potential benefits for spacecraft applications. Among them the possibility to resupply consumables, perform repairs, replace failed components and construct modular structures on orbit. There is a current appeal to build smaller and lighter spacecraft to reduce production time and cost, decrease launch costs, and increase launch availability. Fractionated spacecraft, composed of multiple smaller spacecraft, independently launched and configured in space, may be one way of achieving the benefits of larger satellites with the launch flexibility of small satellites [1]. Control algorithms allowing multiple spacecraft to autonomously avoid each other or rendezvous and dock with each other are a critical component of making autonomous close proximity spacecraft operations.

Previous research and on orbit demonstration of autonomous rendezvous and docking dates back to 1998 [2]. The Air Force Research Laboratory (AFRL) has worked on a series of close-proximity satellite experiments beginning with XSS-10, launched in 2003 to test close-in satellite inspection techniques [3]. Subsequent AFRL programs are under development as described in Refs. [4–7].

NASA tested similar technologies and concepts with the DART mission [8,9].

* Corresponding author. Tel.: +1 518 276 4274.
E-mail addresses: bevilr@rpi.edu (R. Bevilacqua), lehmann_tanya@yahoo.com (T. Lehmann), mromano@nps.edu (M. Romano).

0094-5765/$ - see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.actaastro.2010.08.012

Nomenclature

Abbreviations and acronyms

AFRL	Air Force Research Laboratory
APF	artificial potential function
BCS	body-fixed coordinate system
COM	center of mass
DARPA	Defense Advanced Research Projects Agency
DOF	degrees of freedom
ESA	European Space Agency
ICS	inertial coordinate frame
iGPS	indoor global positioning system
LQR	linear quadratic regulator
LVLH	local vertical local horizontal
NASA	National Aeronautics and Astronautics Administration
NRL	Naval Research Laboratory
NPS	Naval Postgraduate School
PD	proportional-derivative
POSF	Proximity Operations Simulator Facility
RTAI	real-time application interface
SPHERES	synchronized position hold engage and reorient experimental satellites
SRL	Spacecraft Robotics Laboratory

Symbols

$\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$	state space matrices
$\underline{a}$	acceleration determined by LQR/APF control
$\underline{a}_{APF}$	acceleration determined by APF control effort
$\underline{a}_{LQR}$	acceleration determined by LQR control effort
a_m	maximum available acceleration magnitude
$\underline{a}_o$	acceleration of chase spacecraft toward the obstacle
$\underline{a}_{\perp}$	perpendicular acceleration determined by enhanced LQR/APF control
$a_{X,Y,Z}$	acceleration components due to control effort in ICS
$a_{x,y,z}$	acceleration components due to control effort in BCS
D_o	obstacle region of influence
D_{stop}	stopping distance
d_a	decay constant for acceleration toward goal
d_o	factor of safety for obstacle region of influence
$F_{i,req}$	required equivalent force of thruster i
F_t	spacecraft available thrust force
F_θ	force required to affect commanded angular acceleration
J	cost function
$\mathbf{K}_{LQR}$	LQR state feedback gain
k_a	acceleration shaping function
k_s	safety shaping function
k_v	velocity shaping function
L	half the length of the spacecraft simulator measured in the x–y plane
L_{max}	distance between opposite corners of the spacecraft in the x–y plane
L_o	minimum approach distance from chaser COM to obstacle center
L_t	distance from spacecraft simulator COM to each thruster
m_s	mass of spacecraft
$\mathbf{N}$	LQR state-control combination gain matrix
$\mathbf{Q}$	LQR state gain matrix
$\mathbf{R}$	LQR control effort gain matrix
v_m	maximum allowed velocity
$\underline{r}_{ch}$	spacecraft simulator's own position in ICS
$\underline{r}_{dock}$	vector from target COM to the docking port
r_g	chaser's current distance to the goal
r_{init}	chaser's initial distance to the goal
r_m	maximum allowed distance from chaser to goal
r_o	distance between the chaser's COM and the center of the obstacle
$\underline{r}_{obs}$	position of other chase spacecraft simulator in ICS
$\underline{r}_t$	vector from chaser COM to target COM
$\underline{r}_{tg}$	position of the target spacecraft simulator in ICS
$\mathbf{S}$	solution of the algebraic Riccati equation
T_z	torque about the z-axis
u	control effort
$\underline{v}$	relative velocity of chaser spacecraft and obstacle
$\underline{v}_{ch}$	chase simulator's own velocity on the POSF floor in ICS
$\underline{v}_{obs}$	velocity of other chase spacecraft simulator on the POSF floor in ICS
$\underline{v}_{tg}$	velocity of the target spacecraft simulator on the POSF floor in ICS
v_m	maximum allowed relative velocity between spacecraft
$\underline{v}_o$	chaser's velocity toward the center of the obstacle
α_Q	LQR state performance gain
β_R	LQR control effort gain
γ	angle measured counter-clockwise from $\underline{r}_{dock}$ to $\underline{r}_t$
σ	standard deviation for obstacle region of influence
θ_{ch}	chase spacecraft simulator's angular displacement about its z-axis
θ_{tg}	target spacecraft simulator's angular displacement about its z-axis
$\dot{\theta}_{ch}$	chase spacecraft simulator's angular velocity about its z-axis
$\dot{\theta}_{tg}$	target spacecraft simulator's angular velocity about its z-axis
$\ddot{\theta}$	angular acceleration about the z-axis
ω	LVLH angular velocity
x, y, z	LVLH coordinates
$\dot{x}$, $\dot{y}$, $\dot{z}$	LVLH velocities
$\underline{x}=\{x,y,z,\dot{x},\dot{y},\dot{z}\}^T$	state vector

Subscripts and superscripts

$(\cdot)_e$	error

DARPA successfully demonstrated autonomous on-orbit refueling and reconfiguration with the Orbital Express mission in 2007 [10].

ESA's Jules Verne automated transfer vehicle (ATV) tested its anti-collision maneuver system and autonomous rendezvous and docking control system by mating with the International Space Station (ISS) on April 2008 [11]. Artificial potential function control was proposed for the ATV in Ref. [12].

The Massachusetts Institute of Technology (MIT) is currently utilizing the Space Systems Laboratory's synchronized position hold engage and reorient experimental satellites (SPHERES) both on the ground and inside the International Space Station for validation of navigation and control algorithms [13,14].

On the ground experimentation for multiple spacecraft guidance, navigation and control (GNC) has experienced a quick growth in the last years thanks to its low cost and high return.

Additional test beds currently employed include the Jet Propulsion Laboratory's (JPL) formation control test-bed (FCT) [15], the Goddard Space Flight Center's (GSFC) formation flying test-bed (FFTB) [16], and the Naval Research Laboratory's (NRL) front-end robotics enabling near-term demonstration (FREND) [17]. The Field and Space Robotics Laboratory recently proposed experimentation on controlling an incrementally assembled multiple spacecraft system [18].

The Space Robotics Laboratory (SRL), under the Mechanical and Aerospace Engineering Department and the Space Systems Academic Group (SSAG) at the Naval Postgraduate School (NPS), is focused on researching and developing new GNC approaches for multiple spacecraft close-proximity operations. The activities at SRL encompass computer modeling and simulation and on-the-ground experimental validation in the Proximity Operations Simulator Facility (POSF) (see Refs. [19,20,21]). The POSF consists of a 4.9 m × 4.3 m epoxy floor and is used as a platform for various spacecraft simulators with air pads to maneuver with essentially zero friction. In 2007, a new autonomous, distributed control algorithm was developed at NPS for use by multiple spacecraft during close proximity operations, including rendezvous and docking. The control algorithm combines the efficiency of a linear quadratic regulator (LQR) and robust collision avoidance capability of artificial potential function (APF) control. The algorithm underwent preliminarily on-orbit hardware-in-the-loop testing with MIT's SPHERES, through a single spacecraft experiment [22].

The main motivation for this research is to enhance the LQR/APF algorithm to overcome local minima problems. The LQR routine employed for this paper implements, and solves in real time, the complete Riccati problem. Furthermore, the approach is tested for the first time in a multiple vehicle experiment. The main contributions of this work to the state-of-the-art on multiple spacecraft autonomous maneuvering are:

1. A novel autonomous control technique is designed for multiple spacecraft systems. The approach features real-time fuel optimization through the LQR and collision avoidance through the APF. In particular, the base LQR/APF technique is enhanced through a method similar to wall-following in conjunction with high level decisional logic, to overcome local minima issues that would jeopardize the maneuver's success, not mentioning efficiency in terms of duration and fuel consumption.
2. Performance evaluation of the LQR/APF controller is performed through hardware-in-the-loop experimentation involving three vehicles in an assembly maneuver: two chasers and one target.

The paper is organized as follows. Section 2 defines the multiple spacecraft maneuvering problem and the LQR/APF control logic as well as the enhancements achieved by the wall-following technique and decisional

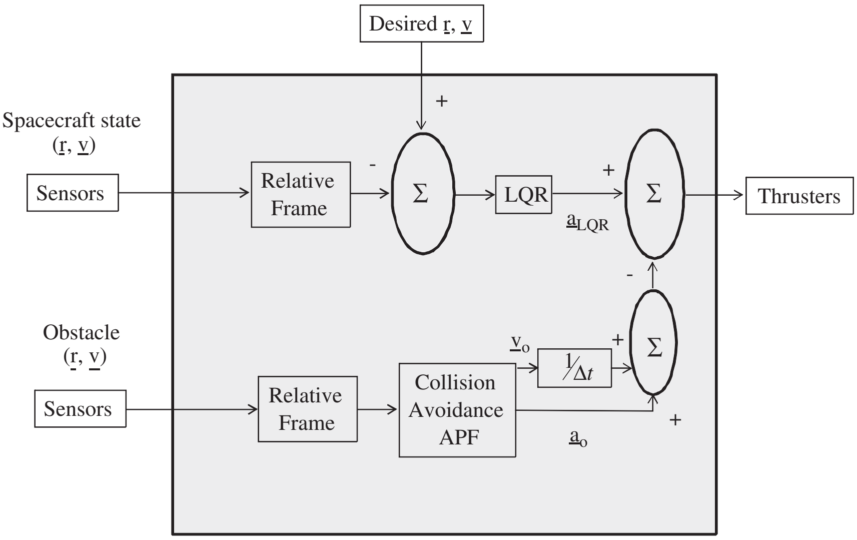

Fig. 1. LQR/APF control block diagram.

logic. Section 3 introduces the spacecraft simulators at the Spacecraft Robotics Laboratory. Section 4 is dedicated to the experimental test involving two chaser spacecraft simulators and one target. Section 5 concludes the paper.

2. LQR/APF control algorithm for multiple spacecraft proximity operations

A feedback controller (Fig. 1) has been previously developed at NPS for spacecraft relative position, and it is the starting point for this work [14]. It combines the efficiency of the linear quadratic regulator (LQR) with the robust collision avoidance capability of the artificial potential function (APF) method. The LQR portion provides the attractive force toward the goal position and the APF portion provides the repulsive effect for collision avoidance. Since the LQR/APF algorithm only provides translational control, it is combined with a quaternion feedback controller for full 6DOF control. The original LQR/APF algorithm was studied and designed for different types of maneuvers [14]. Here we focus on rendezvous and docking. Each spacecraft has an assigned docking side and docking port which has to be driven to match with another spacecraft docking port. Nevertheless, the improvements made to the controller in this work apply to any maneuver within the scope of the original LQR/APF control intended use [14].

2.1. Dynamics of spacecraft relative motion

This section describes the spacecraft relative motion dynamics. The linear dynamics simplifying assumption is necessary to use a linear quadratic regulator, and it is of common use when describing the relative motion of spacecraft in close proximity. The local vertical local horizontal (LVLH) frame can be used to describe the spacecraft relative motion. The LVLH frame is centered at the target spacecraft's center of mass, the *x*-axis points out radially from the center of the Earth to the center of mass of the target, the *y*-axis is aligned in the direction of in-track motion, and the *z*-axis is normal to the orbital plane (Fig. 2). The Clohessy–Wiltshire–Hill [23] linear equations represent a generic chaser's motion relative to a target reference point in a circular orbit, assuming their mutual distance is negligible with respect to the Earth-target distance:

$$\begin{aligned}&\ddot{x}-2\omega\dot{y}-\omega^2x=u_x\\&\ddot{y}+2\omega\dot{x}=u_y\\&\ddot{z}+\omega^2z=u_z\end{aligned}\tag{1}$$

From the dynamics of Eq. (1) the state dynamics matrix **A**, the control mapping matrix **B**, the state-output mapping matrix **C**, and the control-output mapping matrix **D** are the following:

$$\mathbf{A}=\begin{bmatrix}0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\\3\omega^2&0&0&0&2\omega&0\\0&0&0&-2\omega&0&0\\0&0&-\omega^2&0&0&0\end{bmatrix},\quad \mathbf{B}=\begin{bmatrix}0&0&0\\0&0&0\\0&0&0\\1&0&0\\0&1&0\\0&0&1\end{bmatrix}$$

$$\mathbf{C}=I_{6\times6}\mathbf{D}=0_{6\times3}\tag{2}$$

The form of matrix **B** assumes the capability to generate three independent forces in the three directions of space. The forms of **C** and **D** imply complete observability for the state $\underline{x}$.

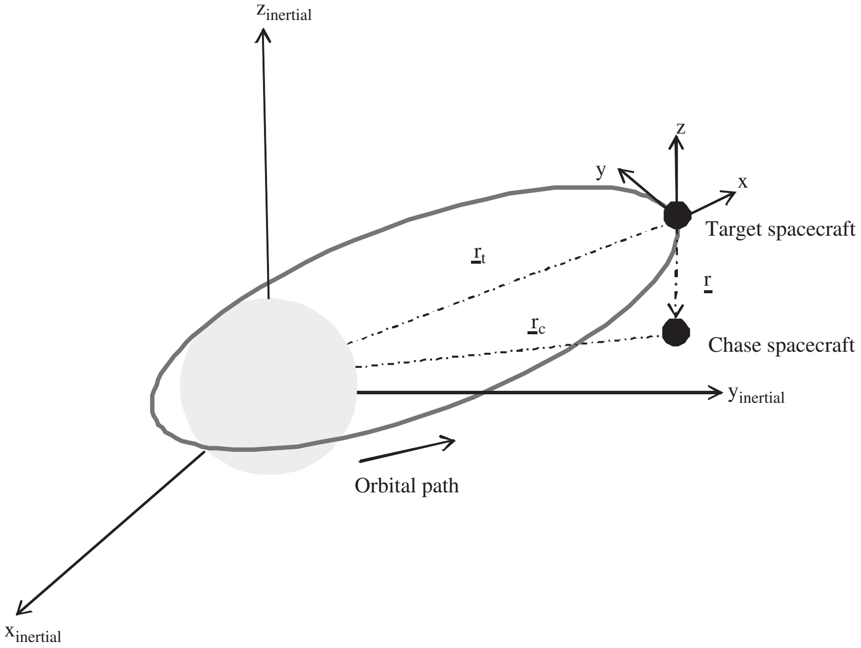

Fig. 2. Relative reference frame.

The attitude control is taken care of by an independent logic and it will not be discussed here.

2.2. LQR control component

The LQR component of the LQR/APF control serves as the attractive force toward the goal position. LQR controllers are based on finding the gain matrix $\mathbf{K_{lqr}}$, generating the optimal control acceleration $\underline{a}_{LQR} = \mathbf{K_{lqr}}\underline{x}_e$ that minimizes the quadratic cost function

$$J = \frac{1}{2}\int_0^{\infty} (\underline{x}_e^T \mathbf{Q}\underline{x}_e + \underline{u}^T \mathbf{R}\underline{u} + 2\underline{u}^T \mathbf{N}\underline{x}_e)\,dt \tag{3}$$

The matrices $\mathbf{Q}$, $\mathbf{R}$, and $\mathbf{N}$ are weighting matrices that allow the designer to balance performance and control effort. The tracking error $\underline{x}_e$ usually refers to a predefined trajectory to be tracked during the maneuver. In this work we do not define a reference trajectory, the LQR control drives the spacecraft toward the goal, so that the tracking error $\underline{x}_e$ is always the difference between the current state vector and the target point. In order to balance the high demand in terms of control effort at the beginning of the maneuver, when $\underline{x}_e$ is high, and the very low forces generated when in the vicinity of the goal, when $\underline{x}_e$ is small, the $\mathbf{Q}$ and $\mathbf{R}$ matrices are iteratively changed as the spacecraft's position changes. The $\mathbf{N}$ matrix is here set to zero. The $\mathbf{Q}$ and $\mathbf{R}$ matrices are as follows:

$$\mathbf{Q} = \begin{bmatrix} \frac{\alpha_{Q_1}}{x_{max}^2} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{\alpha_{Q_2}}{y_{max}^2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\alpha_{Q_3}}{z_{max}^2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\alpha_{Q_4}}{\dot{x}_{max}^2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\alpha_{Q_5}}{\dot{y}_{max}^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{\alpha_{Q_6}}{\dot{z}_{max}^2} \end{bmatrix}$$

$$\mathbf{R} = \begin{bmatrix} \frac{\beta_{R_1}}{u_{x_{max}}^2} & 0 & 0 \\ 0 & \frac{\beta_{R_2}}{u_{y_{max}}^2} & 0 \\ 0 & 0 & \frac{\beta_{R_3}}{u_{z_{max}}^2} \end{bmatrix} \tag{4}$$

The relative position error along each axis is equally weighted in the denominators of the matrix $\mathbf{Q}$ by the square of the distance of the spacecraft's center of mass to the goal position such that

$$x_{max} = y_{max} = z_{max} = r_g \tag{5}$$

The relative velocity error along each axis is weighted such that

$$\dot{x}_{max} = \dot{y}_{max} = \dot{z}_{max} = \frac{r_{init}}{r_m} v_m \tag{6}$$

The numerator terms of the $\mathbf{Q}$ and $\mathbf{R}$ matrices are set equal to the range to the goal such that

$$\alpha_{Q_1} = \alpha_{Q_2} = \alpha_{Q_3} = \alpha_{Q_4} = \alpha_{Q_5} = \alpha_{Q_6} = r_g \tag{7}$$

The denominator terms for the $\mathbf{R}$ matrix are set to the square of the maximum available control

$$u_{x_{max}} = u_{y_{max}} = u_{z_{max}} = a_{z_{max}} = u_m \tag{8}$$

where the thrusters can provide a maximum acceleration along each axis limited to

$$u_m = \frac{F_t}{m_s} \tag{9}$$

and F_t is the thrust force and m_s is the mass of the spacecraft.

2.3. APF control component

The APF portion of the LQR/APF controller uses position-based potential functions to modify the chase spacecraft's velocity as it moves closer to the goal and obstacles. The obstacle can be either foreign objects or another chaser spacecraft, stationary or moving. The target itself is considered as an obstacle irradiating repulsive potential until the chaser is not in the correct relative configuration to complete the docking approach. The repulsive obstacle potential function does not repel the chaser away from obstacles but rather decreases the chaser's velocity and acceleration toward obstacles.

The total control effort due to the combination of LQR and APF is defined as

$$\underline{a} = \underline{a}_{LQR} - \sum_{obs\,=\,0}^{N} \left(\left(k_v \cdot \frac{\underline{v}_o}{\Delta t} \right) + k_s k_a \underline{a}_o \right) \tag{10}$$

where the various parameters are explained in the following.

The velocity shaping parameter k_v is defined as

$$k_v = (e^{-(r_o^2/2\sigma^2)} - e^{-(D_o^2/2\sigma^2)})(e^{-(L_o^2/2\sigma^2)} - e^{-(D_o^2/2\sigma^2)})^{-1} \tag{11}$$

based on the obstacle's distance relative to the chaser, r_o. The parameter k_v is multiplied by the obstacle-chaser relative velocity to ensure that the chaser spacecraft slows to zero at the boundary of the obstacle defined by $r_o = L_o$, the minimum approach distance between the center of mass of the obstacle and center of mass of the chaser. The region of influence of the obstacle, D_o, is calculated by

$$D_o = d_o(L_o + D_{stop}) = (d_o L_0) + \frac{d_o|\underline{v}|^2}{4u_m} \tag{12}$$

Eq. (12) is dependent on the minimum approach distance, L_o, plus the minimum distance in which the chaser spacecraft could stop when applying maximum thrust, D_{stop}. Thus, the region of influence changes with the current relative velocity between the chaser and obstacle. A factor of safety, d_o, is included in Eq. (12) to allow room for the chaser spacecraft to stop gradually, instead of abruptly, to avoid colliding with the obstacle. The standard deviation term, $\sigma = (D_o + L_o)/3$, is applied for the same reason as it causes the chaser spacecraft to reach zero velocity within one standard deviation of the obstacle's boundary.

When the chase spacecraft is within the region of influence of an obstacle, an acceleration shaping function decreases the LQR-commanded acceleration toward the obstacle. The acceleration shaping parameter is

$$k_a = e^{-d_a(r_o - L_o)} \tag{13}$$

where d_a is a positive constant that determines the parameter's rate of decay. The safety function

$$k_s = 1 - e^{-(2d_a r_g)} \tag{14}$$

causes the k_a-imposed decrease in acceleration to decay faster as the chaser approaches the goal. Since the target is also regarded as an obstacle, this safety function allows the chaser spacecraft to approach the target's docking port. Both parameters, k_a and k_s, are multiplied by the component of the LQR commanded acceleration that is directed toward the obstacle

$$\underline{a}_o = \frac{\underline{r}_o \cdot \underline{a}_{LQR}}{r_o} \frac{\underline{r}_o}{r_o} \tag{15}$$

to slow or stop the spacecraft's motion in that direction.

2.4. Enhanced LQR/APF control algorithm

The approach described in the previous section presents an intrinsic limitation due to its design. The combination of the attractive LQR force and the repulsive APF force may result in undesired minima. Since no high level monitoring is performed to prevent conditions such as partial or complete canceling out of the two forces, a local minima may be encountered and the spacecraft can remain trapped in it, especially if the maneuvering vehicle is proceeding toward the goal at a very low approaching rate, typical of safe rendezvous and docking. The proof of this limitation has been previously obtained in simulation and it is finally given here with a direct experimental test, in Section 4.1. Fig. 3 is a representative sketch of a one-chaser one-target in-plane case with an obstacle, where the attractive LQR and the repulsive APF forces cancel each other. Furthermore, Fig. 4 shows a two-chaser one-target in-plane case where the LQR and APF forces also cancel each other. In particular, the situation envisioned in Fig. 4 is taken as test case for the experimental runs reported in Section 4. In order to overcome the above mentioned problems, and make the LQR/APF controller autonomous, the enhancements presented in the following sections have been developed and integrated into the original control logic.

2.4.1. Wall-following-like technique for escaping from obstacles

With the aim of avoiding undesired conditions as in Fig. 3, a technique similar to wall following [24] is added to the pure LQR and APF commands. Obstacles have to be avoided in the fastest and safest way that is possible. To this aim, an in-plane control component, orthogonal and proportional to the APF command, is added to the overall commanded force. The $\underline{a}_\perp$ is only generated in the xy plane of the LVLH reference frame for two main reasons: (a) it is sufficient to avoid local minima problems, (b) usually the spacecraft are in the same orbital plane for the last phases of docking, i.e. $z = \dot{z} = \ddot{z} = 0$. There are two possibilities for the commanded $\underline{a}_\perp$

$$(1)\ \underline{a}_\perp = c\begin{bmatrix} -a_{APF_y} \\ a_{APF_x} \\ 0 \end{bmatrix}, \quad (2)\ \underline{a}_\perp = c\begin{bmatrix} a_{APF_y} \\ -a_{APF_x} \\ 0 \end{bmatrix} \tag{16}$$

The scalar c of Eq. (16) is a positive proportional factor chosen by the control designer. Of the two possibilities the one satisfying Eq. (17) is chosen, i.e. the one driving to

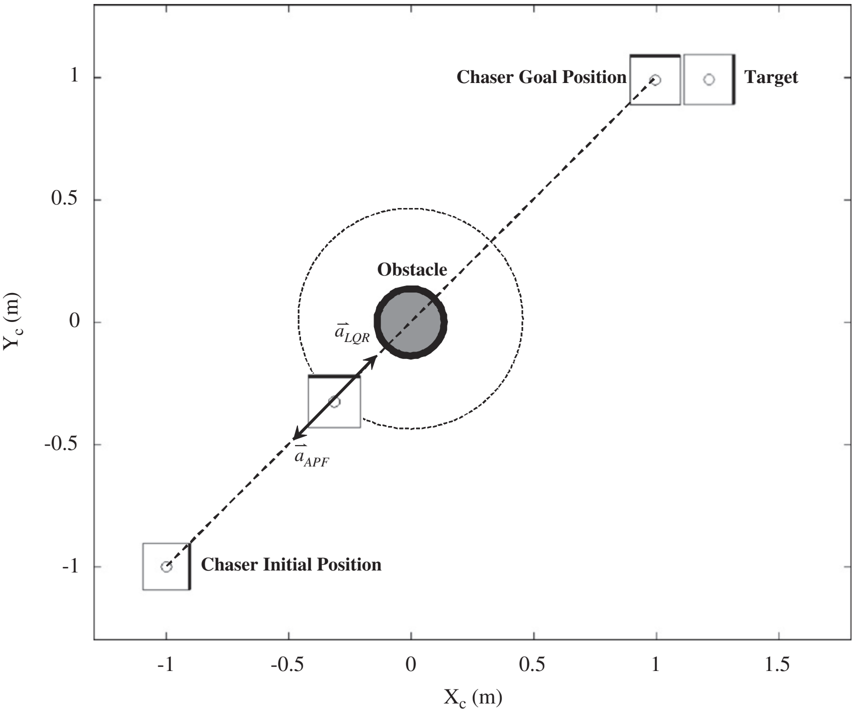

Fig. 3. Representative equilibrium condition between LQR and APF forces in a one-chaser one-target case with obstacle.

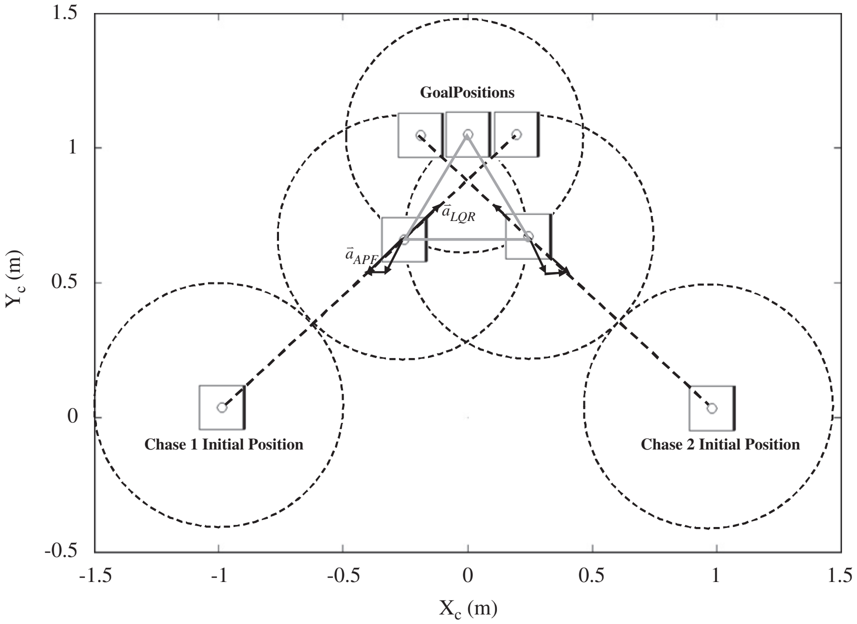

Fig. 4. Representative equilibrium condition between LQR and APF forces in a two-chaser one-target case.

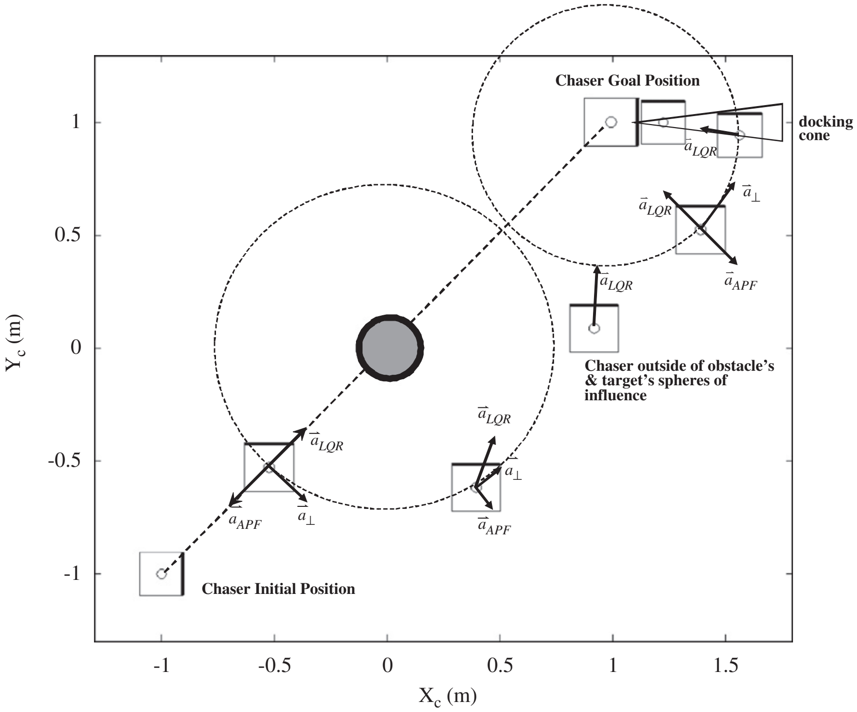

Fig. 5. Representative enhanced LQR/APF logic for obstacle avoidance an target docking.

the target point along the shortest way:

$$a\cos\left(\underline{a}_{\perp}\cdot\frac{\underline{r}_{dock}}{a_{\perp}r_{dock}}\right)<\frac{\pi}{2} \tag{17}$$

Fig. 5 illustrates the modified LQR/APF algorithm with the perpendicular acceleration $\underline{a}_{\perp}$. The acceleration component perpendicular to the APF repulsion allows for safe and guaranteed maneuvering around the obstacle, overcoming the problem described in Fig. 3.

An additional enhancement made to the original LQR/APF control is the introduction of a safety docking cone associated with the docking port on the target spacecraft. As Fig. 5 shows, the APF component is kept constant when the obstacle is the target. The APF control force is turned off only when the center of mass of the chaser spacecraft is within the safety docking cone. In the original LQR/APF [25] the APF repulsive component was decreased as the decreasing distance from the assigned docking position, in order to allow closing between chaser and target. A drawback of this approach is the possibility of collisions especially when the docking port is occluded from the chaser's approaching direction.

Despite these modifications, the control algorithm may still encounter equilibrium conditions when a multiple spacecraft symmetric configuration as the one described in Fig. 4 is encountered. In these situations, the spacecraft are assigned adversarial transversal accelerations to reach the respective goals in the shortest way. For this particular case a higher level logic is introduced, to detect and escape undesired equilibrium conditions in multiple spacecraft maneuvers. The following section describes the additional high level enhancement made to the LQR/APF logic in this sense.

2.4.2. Detection and escaping of local minima in multiple spacecraft maneuvering

Fig. 6 illustrates the proposed technique to avoid local minima when multiple maneuvering spacecraft reach symmetric configurations as in Fig. 4. If the following ''no moving further'' condition occurs during the docking phase, on any of the maneuvering chasers

$$\begin{aligned}&|x(t)-x(t+\Delta t)| < \delta_x\\&|y(t)-y(t+\Delta t)| < \delta_y\\&|z(t)-z(t+\Delta t)| < \delta_z\end{aligned} \tag{18}$$

the transversal accelerations around the target of all the chasers are changed, once and for all, to run the chasers around the target in the same direction, so that, if Eq. (18) is verified for more than one vehicle, they all agree on moving with no additional conflicts around the goal spacecraft (e.g. counterclockwise in the *xy* plane in the example of Fig. 6).

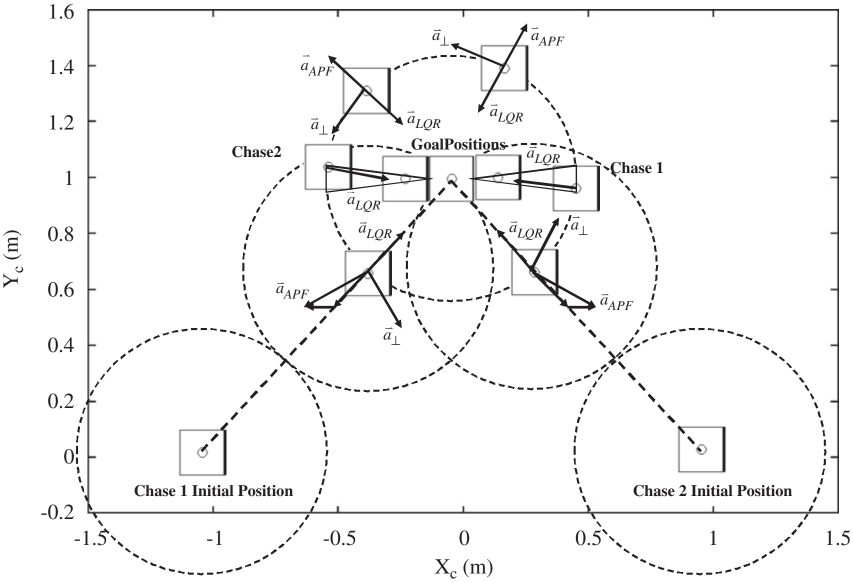

Fig. 6. Representative enhanced LQR/APF logic for multiple spacecraft symmetry problem solution.

2.5. Real-time LQR

An important improvement made to the software with respect to [22] is the employment of a complete real-time LQR solver.

The classical Matlab command "LQR" cannot be compiled into a real-time executable by Real-Time Workshop, a Mathworks tool employed in this research for transferring algorithms onto the real-time hardware [26]. Nevertheless, an LQR Simulink solver (Fig. 7) was available for download at the Mathworks File Exchange website [27] at the moment of beginning this research. This routine solves the complete algebraic Riccati equation accepting the input matrices: **A** (dynamics matrix), **B** (control matrix), **C** (state-output mapping matrix), **D** (control-output mapping matrix), **Q** and **R** weighting matrices. The outputs are the LQR gain matrix, **K**, which is the solution to the associated algebraic Riccati equation, the matrix **S**, and a two dimensional vector, $\underline{E}$, whose first element indicates an error when it is greater than zero or a somewhat unreliable result when it is negative. The second element is the condition number of the **R** matrix. Fig. 7 shows the Simulink block dedicated to

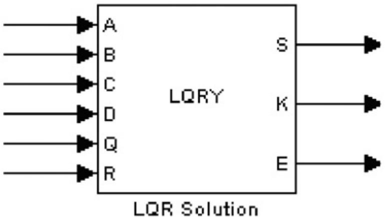

Fig. 7. SIMULINK LQR block.

solving the Riccati problem. At the time of downloading this block, it was not implemented for compilation through Real-Time Workshop, neither for XPCtarget nor for RTAI Linux. The necessary integrations and changes were made within this paper framework, to make the LQR Simulink block compatible with Real-Time Workshop, both for compilation under XPCtarget and RTAI Linux. The updated version of this Simulink LQR solver has been made available by the authors for download [27].

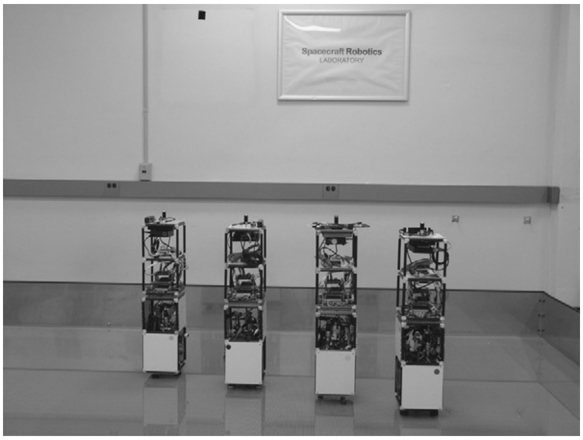

Fig. 8. Multi-spacecraft test-bed at the SRL, US NPS.

3. Hardware-in-the-loop 3rd generation spacecraft simulators

This section introduces the third generation of Spacecraft Simulators developed at the Spacecraft Robotics Laboratory of the US Naval Postgraduate School, and used for the experimental verification of the proposed control method. Fig. 8 shows the fleet of four third-generation operational spacecraft simulators. The simulators float using air pads over a very flat epoxy floor, reproducing a frictionless environment in two dimensions and three degrees of freedom, i.e. two degrees of freedom for the translation and one for the rotation. The actuation is performed via eight body-fixed supersonic thrusters [28]. Two generations of spacecraft simulators were previously developed and used to validate other proximity maneuver

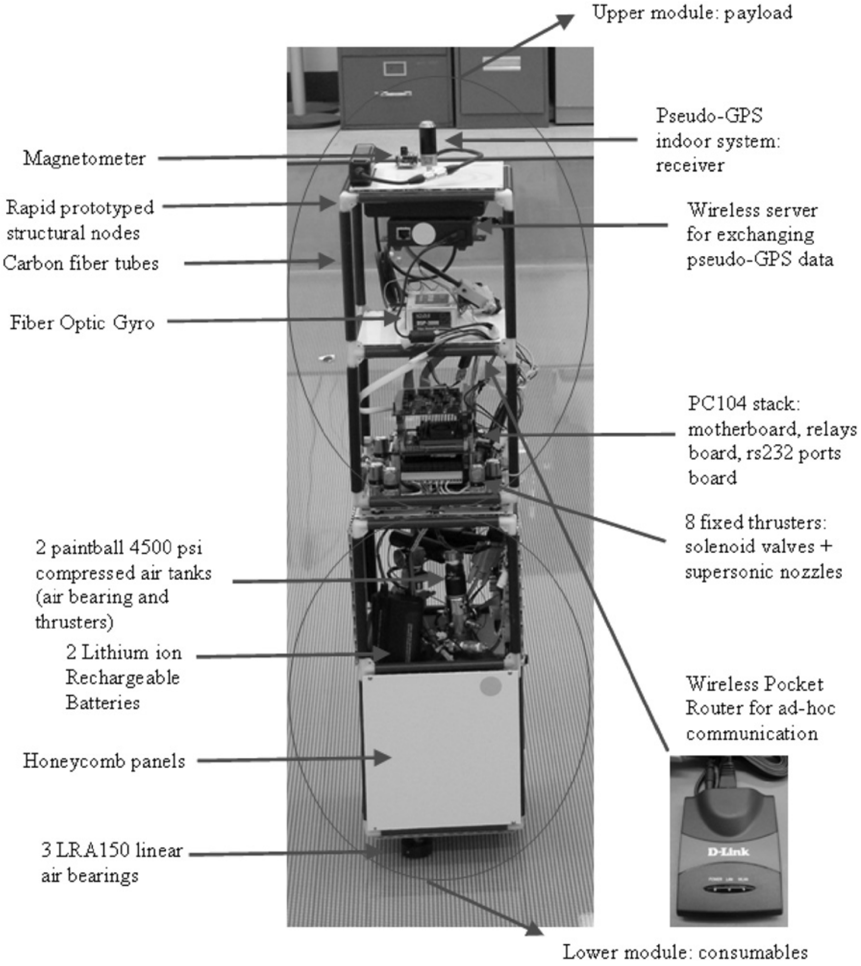
Upper module: payload
Magnetometer
Pseudo-GPS indoor system: receiver
Rapid prototyped structural nodes
Carbon fiber tubes
Wireless server for exchanging pseudo-GPS data
Fiber Optic Gyro
PC104 stack: motherboard, relays board, rs232 ports board
8 fixed thrusters: solenoid valves + supersonic nozzles
2 paintball 4500 psi compressed air tanks (air bearing and thrusters)
2 Lithium ion Rechargeable Batteries
Wireless Pocket Router for ad-hoc communication
Honeycomb panels
3 LRA150 linear air bearings
D-Link
Lower module: consumables

Fig. 9. Detail of the hardware components of a spacecraft simulator.

strategies [19,20]. The reduction of the degrees of freedom from six of the real world to three of the POSF still allows for the verification of the integration between the algorithms and the sensors/actuators, accurately reproducing the dynamics of multi-satellites close proximity flight. Fig. 9 shows the spacecraft simulator's hardware main components and their location on the structure.

The key features of the spacecraft simulators are:

1. *Ad hoc wireless communication*: The robots exchange data among each other and with the external world over the wireless network, with no access point. In this way the system is robust to the loss of one of the simulators.
2. *Modularity*: The simulators are divided into two levels, the payload can be disconnected from the consumables layer and different payload simulators can be used (Fig. 9).
3. Small footprint (20 cm side square).
4. Light weight (10 kg).

In particular the wireless communication real-time capability has been previously experimentally verified by a distributed computing test [21]. Table 1 lists the main components of the electronics used on board each spacecraft simulator.

The wi–fi capability of each robot is also exploited to receive its own absolute position in the laboratory, sensed by the pseudo-GPS indoor system. In particular, the wi–fi communication is currently exploited for relative navigation, by having each robot broadcasting its state vector to the others. The onboard real-time operating system is RTAI Linux [29], in the Debian 2.6.19 version. The classical use of xPC Target by MathWorks as a real-time operating system (OS) is common in academic research [26]. A key advantage of xPC Target is its seamless integration between Simulink via Real-Time Workshop which allows for rapid prototyping of navigation and control algorithms for real-time requirements. Real-Time Workshop automatically generates C code from a Simulink model and the corresponding executable file for a xPC Target based computer. On the other hand, xPC Target has some disadvantages that include support for a limited number of hardware components and no support for USB or Firewire devices. Furthermore, the inaccessibility of its source code, due to its proprietary commercial nature, makes it challenging to add or modify drivers for unsupported hardware.

Table 1
Component electronics hardware description.

Part's name and manufacturer	Description
PC104 (plus) motherboard (advanced digital logic)	SmartCoreT3-400, 400 MHz CPU, SDRAM256-PS
Compact flash (SanDisk extreme IV)	8 Gbyte capacity
20 relays board (IR-104-PBF) (diamond systems)	High density opto-isolated input + relay output
Eight serial ports board (MSMX104+) (advanced digital logic)	–
Firewire PC104 board (embedded designs plus)	IEEE1394 card with 16 bit PC104
Compact wireless-G USB adapter (Linksys)	54 Mbps 802.11 b/g wireless USB network interface adapter
Wireless pocket router/ AP DWL-G730AP (D-Link)	2.4 GHz 802.11 g, ethernet to wireless converter
Solenoid valves (Predyne)	2 way, 24VDC, 2 W
Fiber optic gyro DSP3000 (KVH)	Single axis rate, 100 Hz, asynchronous, RS-232
Magnetometer, MicroMag-3Axis (evaluation kit with RS232 board) (PNI)	Asynchronous, RS-232 (the evaluation kit is still a development version)
DC/DC converters: EK-05 battery controller and regulator + DC1U-1VR 24 V DC/DC converter (ocean server)	3.3, 5, 12, 24 V outputs. The main board is equipped with a batteries' status controller
Battery (inspired energy)	Lithium ion rechargeable battery (95 W h)
Metris iGPS pseudo-GPS indoor system	–

RTAI Linux has been successfully used as an onboard real-time OS. RTAI is a patch to the Linux kernel that allows for the execution of real-time tasks in Linux [30,31]. The RTAI Linux solution is being widely exploited in several engineering areas [32–35]. In this work, we use RTAI Linux with a wide variety of hardware interfaces to include: wireless ad hoc radio communication using UDP, RS232 interface with the sensor suite and power system, and a PC/104 relay board for actuating compressed air nozzles. RTAI Linux also allows for automatic generation of C code from Simulink models through Real-Time Workshop with the executable file for the onboard computers being created outside MATLAB by compilation of the C code.

The details on the ad hoc wireless network and hardware-software interfaces developed for the Spacecraft Simulators are available in Ref. [21].

4. Experimental results

This section reports a two-chaser one target experiment, presenting a symmetric configuration between the chasers' initial conditions and their assigned docking ports on the target (see Fig. 6). In Section 4.1 the classical LQR/APF control algorithm presented in Ref. [14] is tested for the first time in a multiple vehicles experiment, highlighting the limitations of it in case of symmetries and related local minima. Section 4.2 presents the same experiment performed with the enhanced LQR/APF control logic. In the classical LQR/APF case the controller is not capable of driving the maneuver to completion, while the enhanced controller is capable of satisfactorily conclude the experiment, autonomously detecting and overcoming local minima issues.

In order to experimentally test the LQR/APF controllers using the spacecraft simulators, the spacecraft dynamics is reduced to a double integrator. In other words the matrix **A** of Eq. (2) reduces to

$$\mathbf{A}=\begin{bmatrix}0&0&1&0\\0&0&0&1\\0&0&0&0\\0&0&0&0\end{bmatrix} \tag{19}$$

The orbital dynamics can be considered to be a disturbance compensated for by the simulators' navigation and control systems. Since the orbital angular velocity is very small and the very close-proximity maneuvers are completed in much less than one orbital period, this assumption is considered reasonable. As a result, experimentation on the planar floor test bed reproduces many of the critical aspects of actual autonomous proximity operations and can be used to validate computer simulations [36]. Fig. 10 is a block diagram of the LQR/APF control adapted for the NPS simulators. For further details about the single components of the control loop, such as the thruster mapping, the pulse width modulator (PWM), the state estimator (SE), and the PD controller employed for the simulators' orientation, refer to [20]. The need of a thruster mapping algorithm is due to the required conversion of the torque about the z-axis and the forces F_x and F_y generated by the LQR/APF algorithm into commands for the eight fixed thrusters. In particular, the LQR/APF algorithm computes required controls in the LVLH frame x, y, z, which corresponds to the laboratory frame for the experimentation. These controls

Table 2
Experimentation main parameters.

Parameter	Value
L_{max}	0.27 m
L_o	$2L_{max}$
d_o	3 m
u_m	0.016 m/s^2
d_a	1 m
c	0.1
v_m	0.02 m/s
Δt	5 s
δx	3 cm
δ_y	3 cm
δ_z	3 cm
Direction of motion around target for no conflict situation	z counterclockwise

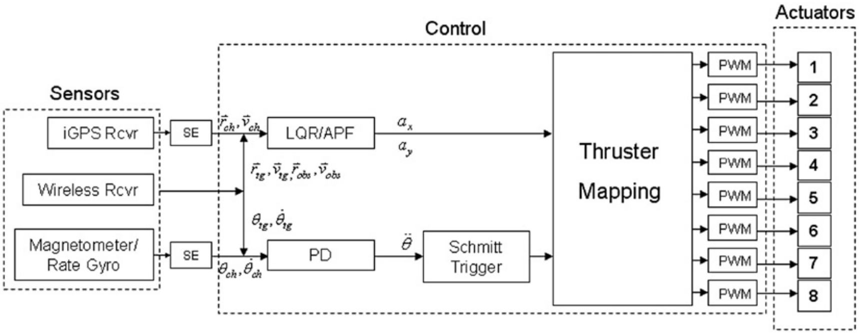

Fig. 10. Block diagram of the control system for the spacecraft simulator.

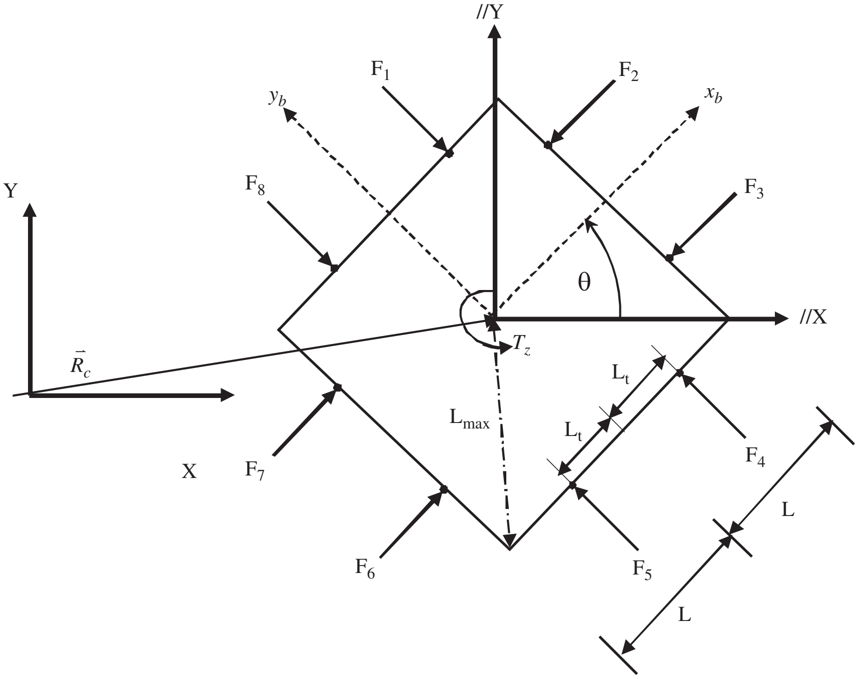

Fig. 11. SRL simulators main reference frames.

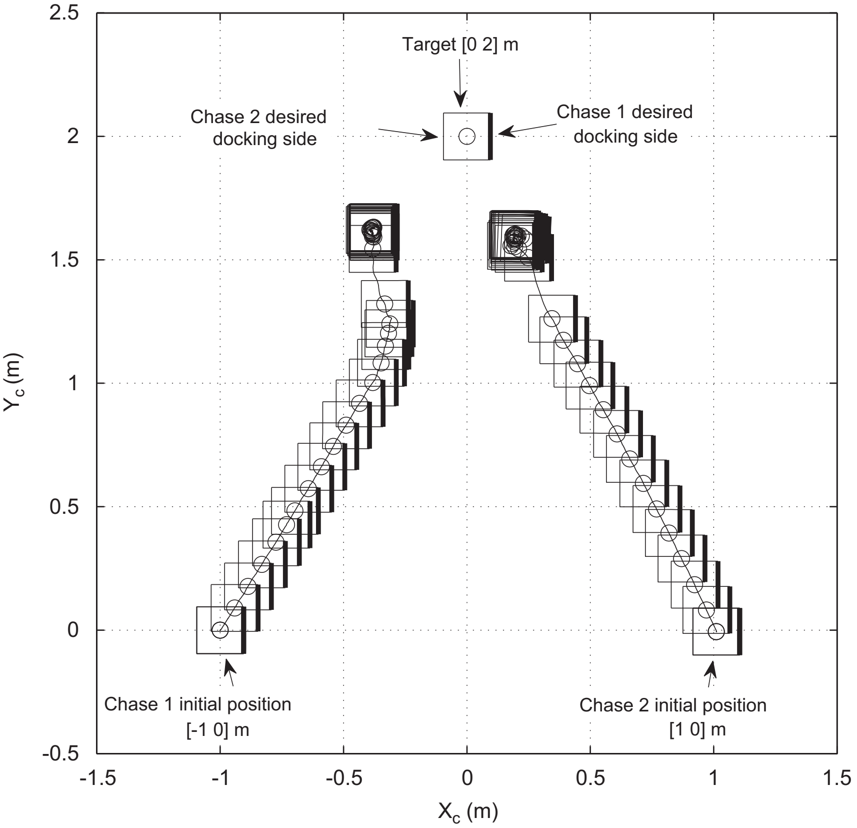

Fig. 12. Experimental result: unsuccessful two-chaser one target symmetric configuration with standard LQR/APF. The bold line is used to visualize the simulators' heading throughout the maneuver. The local minima, illustrated in Fig. 4, are evidently encountered. Movie of this experiment is available at the webpage http://aa.nps.edu/~mromano/pixmovies.htm.

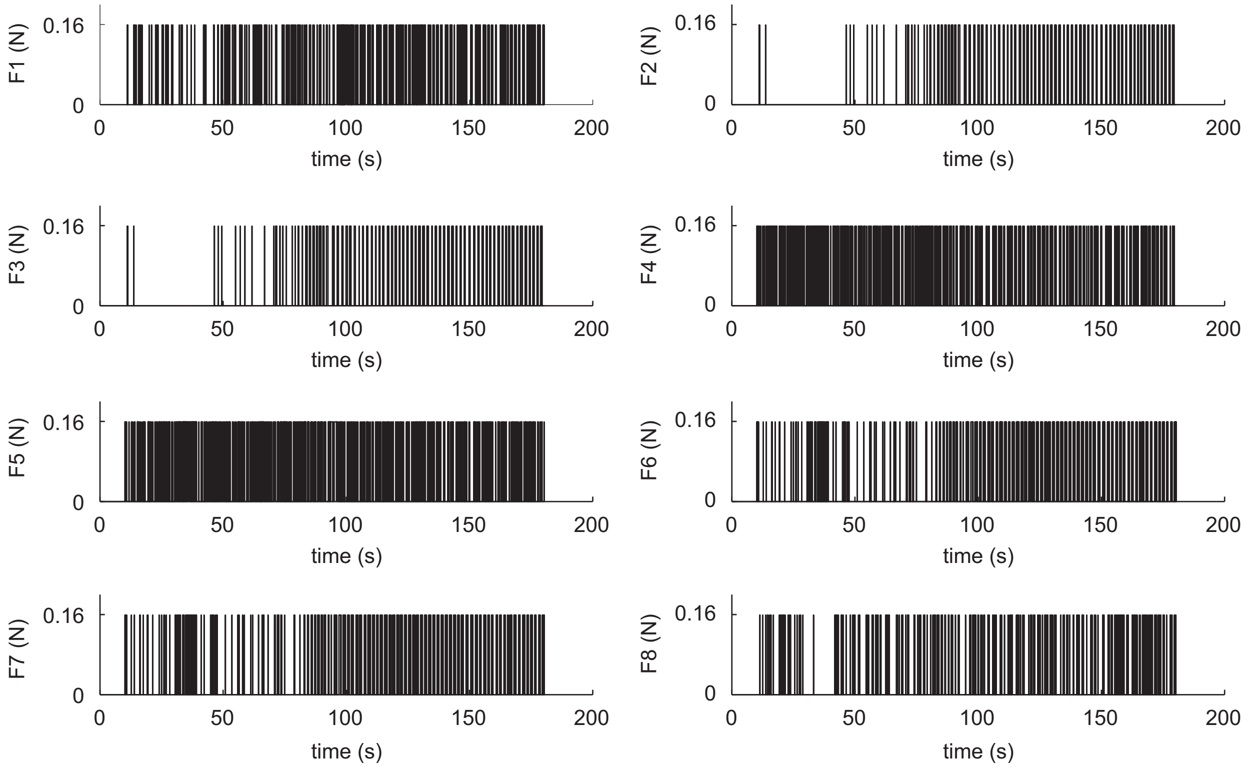

Fig. 13. Experimental result: thrust history for Chaser 1 during test in symmetric configuration with standard LQR/APF. Movie of this experiment is available at the webpage http://aa.nps.edu/~mromano/pixmovies.htm.

are first projected into the simulator's body frame x_b, y_b, z_b, then allocated among the eight actuators [37] (see also Fig. 11).

In both the following experiments the simulators' orientation is maintained at a constant 0° with respect to the laboratory frame, via a PD control. The control algorithm runs at 100 Hz updates, the pulse width modulation filters the required control from the LQR/APF logic at 50 Hz.

The maneuver is considered completed when the distance between target and chaser is below 3 cm.

Table 2 lists the values of control parameters used for the experiments.

4.1. Experiment with the base LQR/APF

Fig. 12 is the bird's eye view of the two chasers one target experiment performed using the unmodified LQR/APF of [22]. As foreseen in Section 2.4 the control design drives the chasers in an equilibrium configuration, where thrusting occurs only to oscillate the robots' positions about the respective local minima. Figs. 13 and 14 represent the on-off history of the eight on board fixed thrusters. The last 60 s of the maneuver are spent with the simulators virtually oscillating about a fixed position, thrusting occurs as commanded by the attractive LQR and repulsive APF, in a equilibrium fashion. Table 3 collects the main parameters evaluating the quality of the maneuver: completion of the maneuver (success or not), required time, and total propellant consumption.

The classical LQR/APF approach results unsuccessful as expected. The next section shows how the enhancement to the control logic lead to a smoother and successful maneuver.

4.2. Experiment with enhanced LQR/APF

Fig. 15 is the bird's eye view of the two chasers one target experiment performed using the enhanced LQR/APF. When the local minima of the classical LQR/APF is encountered and maintained for a time longer than the tolerance Δt, the decisional logic takes over and tasks the counterclockwise motion of the chasers around the target.

As the experimental run shown in Fig. 15 the chasers approach the target only when the field of view constraint is respected, i.e. when the chasers are in the docking cone defined in Fig. 5. Figs. 16 and 17 represent the on-off history of the eight on board fixed thrusters. Table 4 collects the main parameters evaluating the quality of the

Table 3
Symmetric experiment with classical LQR/APF.

Parameter	Value
Maneuver completed	No
Required time (maneuver terminated manually after more than 60 s with no progress)	171 s
Spacecraft 1 and 2 propellant consumption ($\int \underline{u} \cdot \underline{u}\, dt$)	8.39 m/s

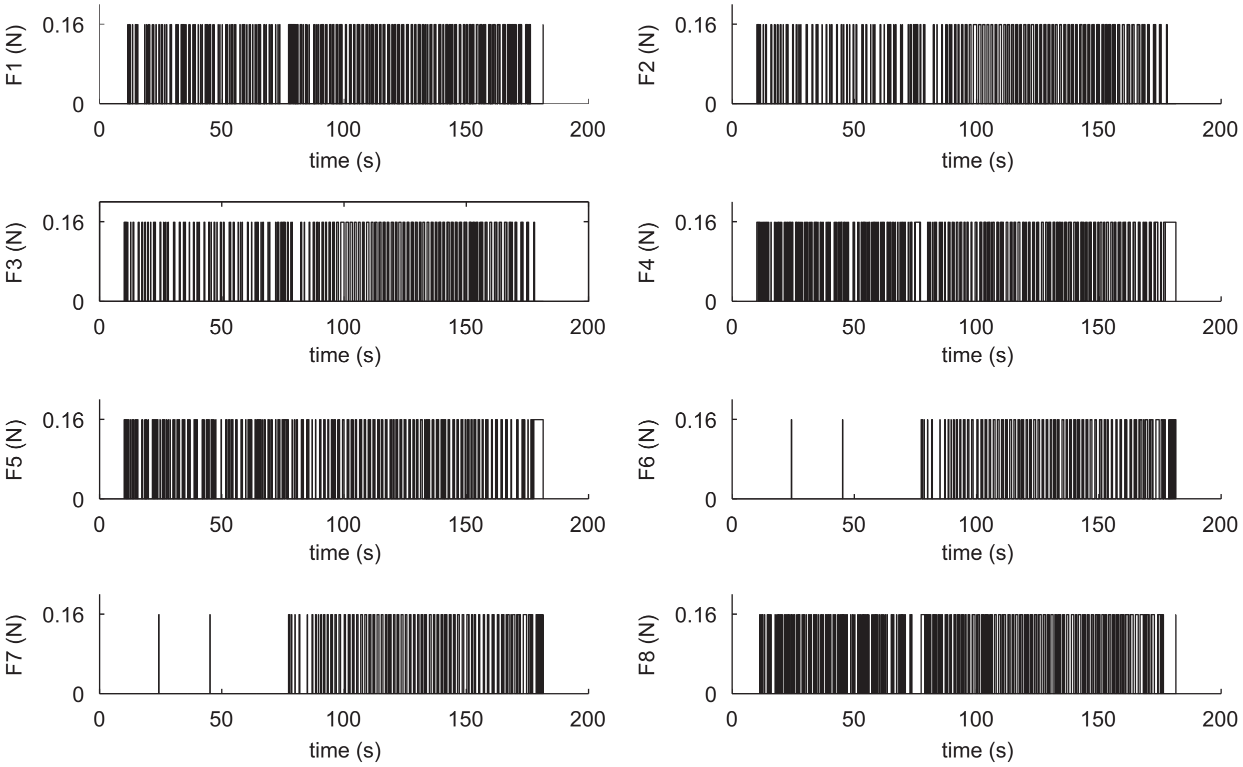

Fig. 14. Experimental result: thrust history for Chaser 2 during test in symmetric configuration with standard LQR/APF. Movie of this experiment is available at the webpage http://aa.nps.edu/∼mromano/pixmovies.htm.

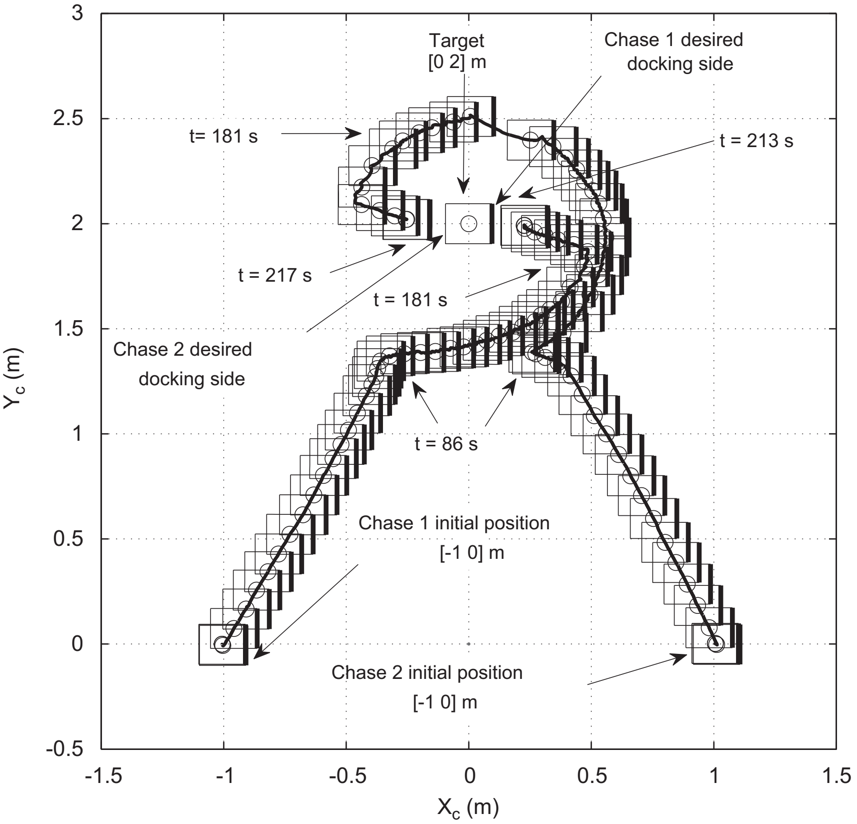

Fig. 15. Experimental result: two-chaser one target symmetric configuration with enhanced LQR/APF. The bold line is used to visualize the simulators' heading throughout the maneuver. Movie of this experiment is available at the webpage http://aa.nps.edu/~mromano/pixmovies.htm.

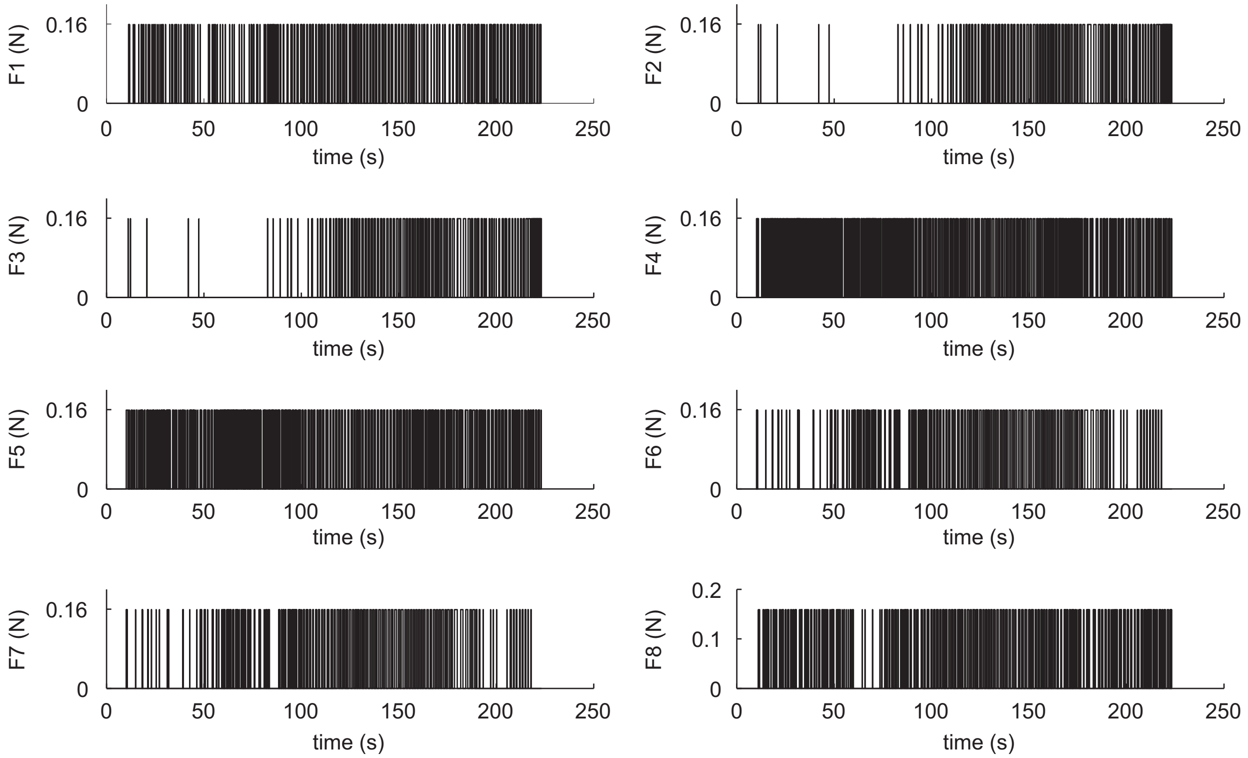

Fig. 16. Experimental result: thrust history for Chaser 1 during test in symmetric configuration with enhanced LQR/APF. Movie of this experiment is available at the webpage http://aa.nps.edu/~mromano/pixmovies.htm.

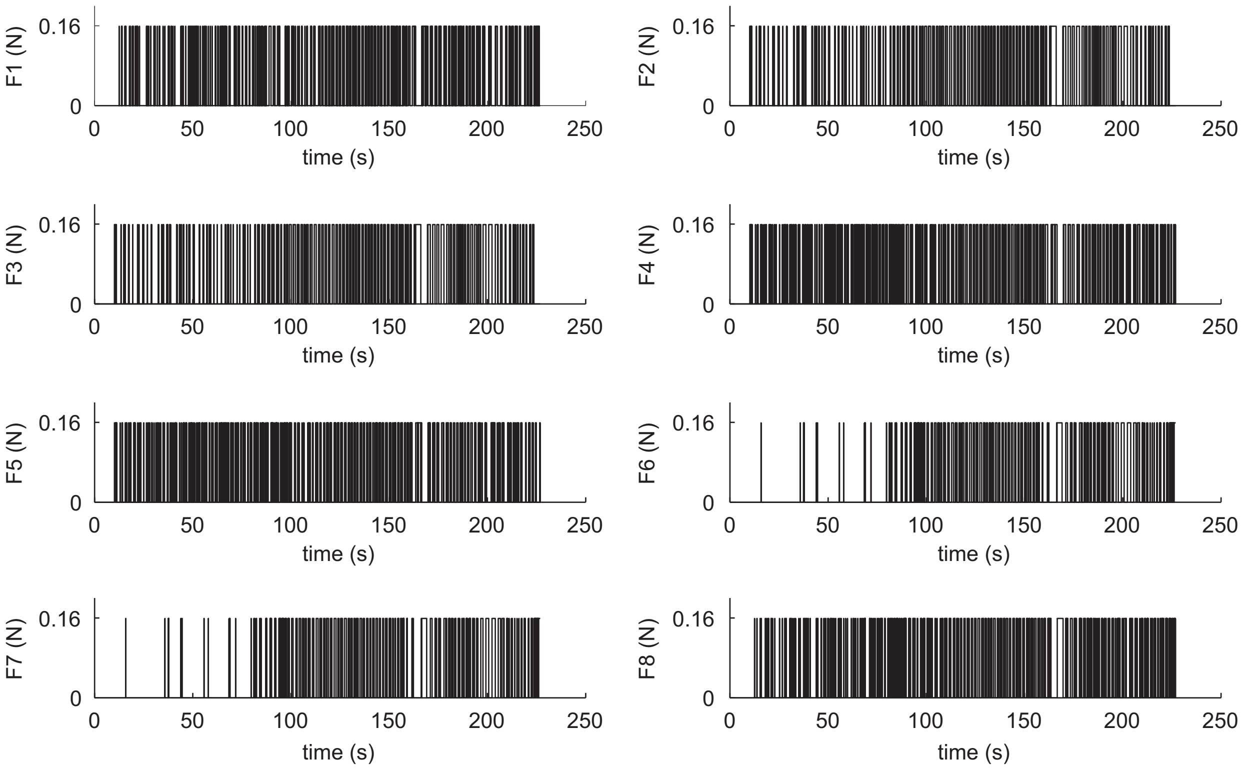

Fig. 17. Experimental result: thrust history for Chaser 2 during test in symmetric configuration with enhanced LQR/APF. Movie of this experiment is available at the webpage http://aa.nps.edu/~mromano/pixmovies.htm.

Table 4
Symmetric experiment with enhanced LQR/APF.

Parameter	Value
Maneuver completed	Yes
Required time	216 s
Spacecraft 1 and 2 propellant consumption ($\int \underline{u} \cdot \underline{u}\, dt$)	12.68 m/s

maneuver: completion of the maneuver (success or not), required time, and total propellant consumption.

5. Conclusion

This work presents a multiple spacecraft control algorithm exploiting linear quadratic regulator, for optimizing the fuel consumption, and an artificial potential function technique, for collision avoidance. By starting from previous developments, integrating the original approach with a technique similar to wall-following and a high level decisional logic, the LQR/APF is enhanced and is made capable of avoiding local minima issues. Furthermore, the here employed LQR routine is a complete solver of the Riccati equation in real time, implemented for execution on RTAI Linux. The designed algorithm is tested, for the first time, in a three spacecraft simulators hardware-in-the-loop experiment, employing the third generation spacecraft simulators of the Spacecraft Robotics Laboratory at the Naval Postgraduate School. Two one-target, two-chaser experiments are presented. The chasers' initial conditions produce local minima in the vicinities of the target, which result in a unresolvable condition for the classical LQR/APF. The same configuration is successfully handled by the enhanced LQR/APF logic, constituting a valid novel approach for autonomous multiple spacecraft maneuvering.

Acknowledgements

This research was partially supported by DARPA. This research was performed while Dr. Bevilacqua was holding a National Research Council Research Associateship Award at the Spacecraft Robotics Laboratory of the US Naval Postgraduate School.

The authors wish to thank LCDR Jason S. Hall for his support on the LQR Simulink block developments.

References

[1] O. Brown, Application of value-centric design to space architectures: the case of fractionated spacecraft, AIAA-2008-7869, AIAA SPACE 2008 Conference and Exposition, San Diego, California, September 9–11, 2008.

[2] I. Kawano, M. Mokuno, T. Kasai, T. Suzuki, Result of autonomous rendezvous and docking experiment of engineering test satellite-VII, Journal of Spacecraft and Rockets 38 (2001) 105–111, doi:10.2514/2.3661.

[3] T. Davis, D. Melanson, XSS-10 micro-satellite flight demonstration program results, in: Proceedings of SPIE Conference on Spacecraft Platforms and Infrastructure, vol. 5419, Bellingham, WA, 2008, pp. 199–203.

[4] I. Mitchell, et al., GNC development of the XSS-11 micro satellite for autonomous rendezvous and proximity operations, in: Proceedings

from the 29th Annual AAS Guidance and Control Conference, Breckenridge, CO, 2006.

[5] D. Shiga, ANGELS to watch over US air force satellites 〈http://space.newscientist.com/article/dn9674〉, 2006.

[6] spaceref.com, Lockheed Martin awarded air force contract for ANGELS nanosatellite program 〈http://www.spaceref.com/news/viewpr.html?pid=20533〉, 2006.

[7] orbital.com, Orbital awarded $29.5 million contract for ANGELS satellite program by Air Force Research Laboratory 〈http://www.orbital.com/NewsInfo/release.asp?prid=635〉, 2007.

[8] nasa.gov, DART demonstrator to test future autonomous rendezvous technologies in orbit 〈www.nasa.gov/centers/marshall/pdf/172361main_dart.pdf〉, 2004.

[9] space.com, Fender Bender: NASA's DART spacecraft bumped into target satellite 〈http://www.space.com/missionlaunches/050422_dart_update.html〉, 2007.

[10] R. Howard, et al., The advanced video guidance sensor: orbital express and the next generation, in: Proceedings of the 2008 Space Technology and Applications International Forum, Albuquerque, New Mexico, 2008.

[11] esa.int, Jules Verne Prepares for ISS Departure 〈http://www.esa.int/SPECIALS/ATV/SEM5BEO4KKF_0.html〉, 2008.

[12] E.St. John-Olcayto, C.R. McInnes, F. Ankersen, Safety-critical autonomous spacecraft control via potential function guidance, in: Proceedings of the 2007 AIAA Infotech at Aerospace Conference and Exhibit, Rohnert Park, CA, 2007.

[13] A. Otero, A. Chen, D. Miller, M. Hilstad, SPHERES: development of an ISS laboratory for formation flight and docking research, in: IEEE Aerospace Conference Proceedings, 2002.

[14] S.B. McCamish, M. Romano, S. Nolet, C.M. Edwards, D.W. Miller, Flight testing of multiple spacecraft control on SPHERES during close proximity operations, AIAA Journal of Spacecraft and Rockets 46 (6) (2009) doi:10.2514/1.43563.

[15] M. Regehr, et al., The formation control testbed, in: IEEE Aerospace Conference Proceedings, Big Sky, MT, 2004.

[16] R. Burns, B. Naasz, D. Gaylor, J. Higinbotham, An environment for hardware-in-the-loop formation navigation and control, AIAA/AAS Astrodynamics Specialist Conference and Exhibit, Providence, RI, 2004.

[17] J. Lennon, C.G. Henshaw, W. Purdy, An architecture for autonomous control of a robotic satellite grappling mission, AIAA Guidance Navigation and Control Conference and Exhibit, Honolulu, HI, 2008.

[18] C. Toglia, D. Kettler, S. Dubowsky, A study of cooperative control of self-assembling robots in space with experimental validation, International Conference on Robotics and Automation, Kobe, Japan, 2009.

[19] M. Romano, D.A. Friedman, T.J. Shay, Laboratory experimentation of autonomous spacecraft approach and docking to a collaborative target, Journal of Spacecraft and Rockets 44 (2007) 164–173, doi:10.2514/1.22092.

[20] J.S. Hall, M. Romano, Novel robotic spacecraft simulator with mini-control moment gyroscopes and rotating thrusters, in: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, Zurich, Switzerland, 2007.

[21] R. Bevilacqua, J. Hall, J. Horning, M. Romano, Ad hoc wireless networking and shared computation for autonomous multi-robot systems, AIAA Journal of Aerospace Computing, Information, and Communication 6 (5) (2009) 328–353, doi:10.2514/1.40734.

[22] S. McCamish, M. Romano, X. Yun, Autonomous distributed control algorithm for multiple spacecraft in close proximity operations, in: AIAA Guidance, Navigation and Control Conference and Exhibit, Hilton Head, SC, 2007.

[23] W.H. Clohessy, R.S. Wiltshire, Terminal guidance system for satellite rendezvous, Journal of the Aerospace Sciences 27 (9) (1960) 653–658.

[24] X. Yun, K.-C. Tan, A wall-following method for escaping local minima in potential field based motion planning, in: IEEE Eighth International Conference on Advanced Robotics, Monterey, California, 1997, pp. 421–426.

[25] S. McCamish, Distributed autonomous control of multiple spacecraft during close proximity operations, Ph.D. Dissertation, Department of Electrical and Computer Engineering, Naval Postgraduate School, Monterey, California, 2007.

[26] mathworks.com 〈http://www.mathworks.com/products/xpctarget/〉, 2009.

[27] mathworks.com, Mathworks exchange files website 〈http://www.mathworks.com/matlabcentral/fileexchange/2651〉, 2009.

[28] C. Lugini, M. Romano, A ballistic pendulum test stand to characterize small cold-gas thruster nozzles, Acta Astronautica 64 (5–6) (2009) 615–625, doi:10.1016/j.actaastro.2008.08.001.

[29] rtai.org 〈https://www.rtai.org/RTAILAB/RTAI-Lab-tutorial.pdf〉, 2009.

[30] L. Dozio, P. Mantegazza, Real time distributed control system using RTAI, in: Sixth IEE International Symposium on Object-Oriented Real-Time Distributed Computing, Hakodate, Hokkaido, Japan, 2003.

[31] G. Quaranta, P. Mantegazza, Using MATLAB-Simulink RTW to build real time control applications in user space with RTAI-LXRT, Realtime Linux Workshop, Milan, Italy, 2001.

[32] S. Ricci, A. Scotti, Aeroelastic testing on a three surface airplane, in: 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Newport, Rhode Island, 2006.

[33] L. Dozio, A. Toso, W. Corbetta, E. Vigoni, G. Ghiringhelli, Comparison of feedback and feedforward strategies for active structural–acoustic control of broadband sound transmission into a cavity, in: 13th AIAA/CEAS Aeroacoustics Conference (28th AIAA Aeroacoustics Conference), Rome, Italy, 2007.

[34] M.G. Perhinschi, M.R. Napolitano, G. Campa, B. Seanor, S. Gururajan, G. Yu, Design and flight testing of intelligent flight control laws for the WVU YF-22 model aircraft, in: AIAA Guidance, Navigation, and Control Conference and Exhibit, San Francisco, California, 2005.

[35] Y. Zheng, B.J. Martin, N. Villaume, VERSE—virtual equivalent real-time simulation environment, in: AIAA Modeling and Simulation Technologies Conference and Exhibit, San Francisco, California, 2005.

[36] J.S. Hall, M. Romano, Robotic testbed for the experimentation of guidance and control of spacecraft during proximity maneuvers, Mechatronic Systems, In-Tech, Vienna, Austria, ISBN 978-953-7619-46-6.

[37] T. Lehmann, Hardware-in-the-loop testing of distributed autonomous control algorithms for multiple spacecraft close proximity operations using the autonomous Multi-Agent Physically Interacting Spacecraft (AMPHIS) test bed, M.Sc. Dissertation, Department of Mechanical and Astronautical Engineering, Naval Postgraduate School, Monterey, California, 2009.