Stability analysis for the Null-Space-based Behavioral control for multi-robot systems

Gianluca Antonelli   Filippo Arrichiello   Stefano Chiaverini

Abstract— A wide number of mobile multi-robot systems makes use of behavior-based approaches to accomplish their missions. However, despite the advantages in term of flexibility and versatility, the behavior-based approaches often lack of a rigorous stability analysis; when the latter is present it is only valid for specific missions. A behavior based-approach for the control of different robotic systems, namely the Null-Space-based Behavioral control, has been recently proposed. In this paper, its general stability analysis is discussed following a Lyapunov-based approach; moreover, effective conditions are given to verify that the behaviors of specific missions are properly defined and merged. Finally, the stability of several missions for multi-robot systems is discussed.

I. Introduction

In the last decade, multi-robot systems have been object of a widespread research interest and they still pose challenging control problems. Such systems, in fact, need to process many data in real-time, eventually working in an unstructured environment, while accomplishing several tasks such as manipulation, exploration, mapping or moving through predefined via-points. Obviously, the robots of the team also need to avoid static or dynamic obstacles, perform fault detection algorithms and preserve their integrity.

Concerning multi-robot systems' motion control, a wide number of approaches have been presented in the literature and a classification can be done distinguishing between behavior-based approaches and analytical approaches. Many applications such as surveillance robotics [6], inspection [15], rescue robotics [19] are solved by resorting to behavioral approaches since they require to achieve several goals at the same time. Behavior-based approaches, in fact, give the system the autonomy to navigate in complex environments avoiding off-line path planning, using sensors to obtain instantaneous information about the environment and increasing flexibility of the system. Thus, they result useful to guide a robotic system in an unknown or dynamically changing environment. Among the behavioral approaches, seminal works are reported in the papers [10] and [5], while, lately, behavioral approaches have been applied to the formation control of multi-robot systems as in, e.g., [18], [17] and [8]. On the other side, analytical approaches, as e.g. [23], [21], [9], are characterized by a solid stability analysis but suffer from the absence of the appealing feature of versatility proper of behavioral approaches.

Despite the several advantages over analytical approaches, most of the behavior-based approaches presented in the literature lacks of a general procedure for the stability analysis; for most of these approaches, in fact, the stability analysis is strictly referred to the specific mission and has not general validity. Recently, a behavior-based approach for the control of different robotic systems, namely the Null-Space-based Behavioral control (NSB), has been presented (see [4], [3]) and tested with different mobile multi-robot systems performing variegated formation control missions (see [1], [2]). In this paper, the NSB general stability analysis is discussed following a Lyapunov-based approach; moreover, effective conditions to verify that the behaviors of a generic mission are properly defined are given. Applying these results, the stability of several missions for multi-robot systems is finally discussed.

II. NSB Mathematical Background

Generally, a mission involving several robots may requires the accomplishment of several tasks at the same time. A common approach is to decompose the overall mission of the system in elementary tasks (or behaviors), solve them as they were working alone and, finally, combine the outputs of the single tasks to obtain the motion command to each robot. As discussed in [3] the Null-Space-based Behavioral control differs from the other existing methods in the behavioral coordination method, i.e., in the way the outputs of the single elementary behaviors are assembled to compose the final behavior. In particular, the NSB uses a geometric, priority-based composition of the tasks' outputs to obtain the motion reference commands for the robots that allows the system to exhibit robustness with respect to eventually conflicting tasks. The NSB approach has been implemented to generic robotic systems [3] and, in the following, its particularization to a platoon of n robots considered as material point on a planar surface (each robot has 2 degrees of freedom) is recalled.

Let us define as $\boldsymbol{\sigma} \in \mathbb{R}^m$ a proper task variable to be controlled, as $\boldsymbol{p}_i \in \mathbb{R}^2$ the position of the $i-$robot and as $\boldsymbol{p} \in \mathbb{R}^{2n}$ the system configuration, then:

$$\boldsymbol{\sigma} = \boldsymbol{f}(\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n) = \boldsymbol{f}(\boldsymbol{p}) \qquad (1)$$

with the corresponding differential relationship:

$$\dot{\boldsymbol{\sigma}} = \frac{\partial \boldsymbol{f}(\boldsymbol{p})}{\partial \boldsymbol{p}} \boldsymbol{v} = \boldsymbol{J}(\boldsymbol{p})\boldsymbol{v}, \qquad (2)$$

where $\boldsymbol{J}(\boldsymbol{p}) \in \mathbb{R}^{m \times 2n}$ is the configuration-dependent task Jacobian matrix, and $\boldsymbol{v} \in \mathbb{R}^{2n}$ is the system velocity.

Following a singularity-robust task-priority inverse kinematics technique inherited by industrial manipulators applications, the motion references $p_{\text{des}}(t)$ for the system, starting from desired values $\sigma_{\text{des}}(t)$ of the task function, are generated by inverting the (locally linear) mapping in eq. (2)(see [20]). A typical requirement is to pursue minimum-norm velocity, leading to the least-squares solution (dependencies of the Jacobian are dropped out to increase readability):

$$v_{\text{des}} = J^\dagger \dot{\sigma}_{\text{des}} = J^{\text{T}} \left(J J^{\text{T}} \right)^{-1} \dot{\sigma}_{\text{des}}. \qquad (3)$$

In order to avoid the well known problem of numerical drift, a Closed Loop Inverse Kinematics (CLIK) version of the algorithm is usually implemented (see [11]) as

$$v_{\text{des}} = J^\dagger \left(\dot{\sigma}_{\text{des}} + \Lambda \tilde{\sigma} \right), \qquad (4)$$

where Λ is a suitable constant positive-definite matrix of gains and $\tilde{\sigma}$ is the task error defined as

$$\tilde{\sigma} = \sigma_{\text{des}} - \sigma.$$

In case of system redundancy, i.e., if $2n > m$, the classic general solution contains a null projector operator (see [14]):

$$v_{\text{des}} = J^\dagger \left(\dot{\sigma}_{\text{des}} + \Lambda \tilde{\sigma} \right) + \left(I_{2n} - J^\dagger J \right) v_{\text{null}}, \qquad (5)$$

where I_{2n} is the $(2n \times 2n)$ Identity matrix and the vector $v_{\text{null}} \in \mathbb{R}^{2n}$ is an arbitrary system velocity vector. It can be recognized that the operator $\left(I_{2n} - J^\dagger J \right)$ projects a generic velocity vector in the null space of the Jacobian matrix, so as to delete the velocity components that interact with the task.

For highly redundant systems, multiple tasks can be arranged in priority in order to try to fulfill most of them, hopefully all of them, simultaneously. Let us consider, for sake of simplicity, 3 tasks, that will be denoted with the subscript a, b and c, respectively:

$$\sigma_a = f_a(p); \quad \sigma_b = f_b(p); \quad \sigma_c = f_c(p) \qquad (6)$$

where $\sigma_a \in \mathbb{R}^{m_a}$, $\sigma_b \in \mathbb{R}^{m_b}$ and $\sigma_c \in \mathbb{R}^{m_c}$. For each of the tasks, the corresponding Jacobian matrix is properly elaborated as $J_a \in \mathbb{R}^{m_a \times 2n}$, $J_b \in \mathbb{R}^{m_b \times 2n}$ and $J_c \in \mathbb{R}^{m_c \times 2n}$. Let us further define the corresponding null space projectors as

$$N_a = \left(I_{2n} - J_a^\dagger J_a \right)$$
$$N_b = \left(I_{2n} - J_b^\dagger J_b \right).$$

A generalization of the singularity-robust task priority inverse kinematic solution proposed in [11] leads to (see [3]):

$$\begin{aligned} v_{\text{des}} &= J_a^\dagger \Lambda_a \tilde{\sigma}_a + N_a \left(J_b^\dagger \Lambda_b \tilde{\sigma}_b + N_b J_c^\dagger \Lambda_c \tilde{\sigma}_c \right) (7) \\ &= J_a^\dagger \Lambda_a \tilde{\sigma}_a + N_a J_b^\dagger \Lambda_b \tilde{\sigma}_b + N_a N_b J_c^\dagger \Lambda_c \tilde{\sigma}_c \end{aligned}$$

where a regulation problem ($\dot{\sigma}_{a,des} \equiv \dot{\sigma}_{b,des} \equiv \dot{\sigma}_{c,des} \equiv 0$) has been considered and the priority of the tasks follows the alphabetical order. This algorithm has a clear geometrical

interpretation: the tasks are separately inverted by the use of the pseudoinverse of the corresponding Jacobian; the velocities associated with the lower priority task are further projected in the null space of the sole higher task.

However, a *correct* projection is considered in [7], [16] where the generic task is not projected onto the null space of the sole higher priority task but onto the null space of the task achieved by considering the augmented Jacobian of all the higher priority ones. For the 3 tasks example, thus, by defining:

$$J_{ab} = \begin{bmatrix} J_a \\ J_b \end{bmatrix}, \quad N_{ab} = \left(I_{2n} - J_{ab}^\dagger J_{ab} \right) \qquad (8)$$

the desired velocities are

$$v_{\text{des}} = J_a^\dagger \Lambda_a \tilde{\sigma}_a + N_a J_b^\dagger \Lambda_b \tilde{\sigma}_b + N_{ab} J_c^\dagger \Lambda_c \tilde{\sigma}_c \qquad (9)$$

It is worth noticing that the solution in (9) looses the geometrical interpretation of the solution (7) and strongly couples all the tasks. On the other side, the approach in (7) may lead to undesired behavior. It is of interest, thus, to understand the stability of the two solutions to better design the inverse kinematic solution.

A. Definitions

Applying some basic geometric similarities [12], some definitions concerning the relationships between two tasks will be given in this section.

Given two generic tasks, denoted with the lower scripts x and y, they will be defined as *dependent* if:

$$\rho(J_x^\dagger) + \rho(J_y^\dagger) > \rho \left(\begin{bmatrix} J_x^\dagger & J_y^\dagger \end{bmatrix} \right). \qquad (10)$$

where $\rho(\cdot)$ denotes the rank of the matrix. The two tasks will be defined as *independent* if:

$$\rho(J_x^\dagger) + \rho(J_y^\dagger) = \rho \left(\begin{bmatrix} J_x^\dagger & J_y^\dagger \end{bmatrix} \right). \qquad (11)$$

Finally, they will be defined as *orthogonal* if:

$$J_x J_y^\dagger = O_{m_x \times m_y} \qquad (12)$$

where $O_{m_x \times m_y}$ is the $(m_x \times m_y)$ null matrix.

It is worth noticing that the three given conditions of dependency, independency and orthogonality may be verified by resorting to the transpose of the corresponding Jacobians instead of the pseudoinverse; it is known, in fact, that they share the same span. The independency condition, thus, becomes:

$$\rho(J_x^{\text{T}}) + \rho(J_y^{\text{T}}) = \rho \left(\begin{bmatrix} J_x^{\text{T}} & J_y^{\text{T}} \end{bmatrix} \right). \qquad (13)$$

In the following section it will be demonstrated that these definitions, and condition (13), play an important role in the eventual convergence of the task errors. Moreover, condition (13) can be easily verified on the symbolic definition of the Jacobians; it is not necessary, thus, to resort to numerical investigation of the matrices.

III. STABILITY ANALYSIS

For sake of simplicity the stability analysis will be first discussed with respect to sole three tasks, its generalization to a generic number of tasks will then be provided.

Let us define $\tilde{\boldsymbol{\sigma}} \in \mathbb{R}^{m_a+m_b+m_c}$ as

$$\tilde{\boldsymbol{\sigma}} = [\,\tilde{\boldsymbol{\sigma}}_a^{\mathrm{T}} \quad \tilde{\boldsymbol{\sigma}}_b^{\mathrm{T}} \quad \tilde{\boldsymbol{\sigma}}_c^{\mathrm{T}}\,]^{\mathrm{T}}, \tag{14}$$

i.e., the stacked vector of tasks' errors. A possible Lyapunov function candidate is given by

$$V = \frac{1}{2}\tilde{\boldsymbol{\sigma}}^{\mathrm{T}}\tilde{\boldsymbol{\sigma}} \tag{15}$$

whose time derivative, assuming a regulation problem, is

$$\dot{V} = -\tilde{\boldsymbol{\sigma}}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{J}_a \\ \boldsymbol{J}_b \\ \boldsymbol{J}_c \end{bmatrix} \boldsymbol{v} \tag{16}$$

that, substituting the system velocity (7) or (9) into (16), can be rearranged as

$$
\begin{aligned}
\dot{V} &= -\tilde{\boldsymbol{\sigma}}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{\Lambda}_a & \boldsymbol{O}_{m_a,m_b} & \boldsymbol{O}_{m_a,m_c} \\ \boldsymbol{J}_b\boldsymbol{J}_a^{\dagger}\boldsymbol{\Lambda}_a & \boldsymbol{J}_b\boldsymbol{N}_a\boldsymbol{J}_b^{\dagger}\boldsymbol{\Lambda}_b & \boldsymbol{J}_b\overline{\boldsymbol{N}}\boldsymbol{J}_c^{\dagger}\boldsymbol{\Lambda}_c \\ \boldsymbol{J}_c\boldsymbol{J}_a^{\dagger}\boldsymbol{\Lambda}_a & \boldsymbol{J}_c\boldsymbol{N}_a\boldsymbol{J}_b^{\dagger}\boldsymbol{\Lambda}_b & \boldsymbol{J}_c\overline{\boldsymbol{N}}\boldsymbol{J}_c^{\dagger}\boldsymbol{\Lambda}_c \end{bmatrix} \tilde{\boldsymbol{\sigma}} \\
&= -\tilde{\boldsymbol{\sigma}}^{\mathrm{T}} \begin{bmatrix} \boldsymbol{M}_{11} & \boldsymbol{O}_{m_a,m_b} & \boldsymbol{O}_{m_a,m_c} \\ \boldsymbol{M}_{21} & \boldsymbol{M}_{22} & \boldsymbol{M}_{23} \\ \boldsymbol{M}_{31} & \boldsymbol{M}_{32} & \boldsymbol{M}_{33} \end{bmatrix} \tilde{\boldsymbol{\sigma}} \\
&= -\tilde{\boldsymbol{\sigma}}^{\mathrm{T}}\boldsymbol{M}\tilde{\boldsymbol{\sigma}} \tag{17}
\end{aligned}
$$

where

$$\overline{\boldsymbol{N}} = \begin{cases} \boldsymbol{N}_a\boldsymbol{N}_b & \text{for eq. (7)} \\ \boldsymbol{N}_{ab} & \text{for eq. (9).} \end{cases}$$

The sign of $\dot{V}$ in (17) is not generally determined, and the following stability analysis is aimed at founding conditions under which the matrix $\boldsymbol{M}$ results to be positive definite.

The matrix $\boldsymbol{M}$ were decomposed into sub-matrices $\boldsymbol{M}_{ij}$ of proper dimensions. In general, all the sub-matrices are different from the null matrix except for the elements corresponding to the first m_a rows and column ranging from $m_a + 1$ to $m_a + m_b + m_c$, i.e., the sub-matrices $\boldsymbol{M}_{12}$ and $\boldsymbol{M}_{13}$, that are null by construction. A necessary condition for $\boldsymbol{M}$ to be positive definite is that all the sub-matrices on the main diagonal are positive definite. The $(m_a \times m_a)$ sub-matrix $\boldsymbol{M}_{11}$ is obviously positive definite as long as the gain matrix $\boldsymbol{\Lambda}_a > \boldsymbol{0}$. The $(m_b \times m_b)$ sub-matrix $\boldsymbol{M}_{22}$ is positive definite if tasks a and b are independent, i.e., if condition (13) holds and if the gain matrix $\boldsymbol{\Lambda}_b > \boldsymbol{0}$. The $(m_c \times m_c)$ sub-matrix $\boldsymbol{M}_{33}$ is positive definite for eq. (9) if the task c is independent to the augmented Jacobian obtained by stacking tasks a and b and if the gain matrix $\boldsymbol{\Lambda}_c > \boldsymbol{0}$. For eq. (7) the sign of this sub-matrix also depends on the angle among the subspaces, and the independency is not sufficient to prove its positive definitiveness; however, a sufficient condition is that an orthogonality relationship between two of the three tasks holds.

A sufficient condition for $\boldsymbol{M}$ to be positive definite is given by its eventual lower triangular form, thus, it is of interest to verify this condition.

The sign of the sub-matrices holding to the lower triangle will not be determinant for the overall identification of the sign of $\boldsymbol{M}$. For sake of completeness, however, it is worth noticing that the sub-matrices $\boldsymbol{M}_{21}$ and $\boldsymbol{M}_{31}$ are null if the tasks b and c, respectively, and a are orthogonal, otherwise they are not determined in sign. The sub-matrix $\boldsymbol{M}_{32}$ is not null if c and b are independent in the null of a, otherwise it is not determined in sign.

The sub-matrix $\boldsymbol{M}_{23}$ with $\overline{\boldsymbol{N}} = \boldsymbol{N}_{ab}$, i.e., for eq. (9), is always null by construction since the first matrix multiplication $\boldsymbol{J}_b\boldsymbol{N}_{ab} = \boldsymbol{O}$. The use of eq. (7), i.e., with $\overline{\boldsymbol{N}} = \boldsymbol{N}_a\boldsymbol{N}_b$ does not guarantee that this sub-matrix is null, in particular, $\boldsymbol{M}_{23} = \boldsymbol{O}$ only if two successive tasks are orthogonal one each other, i.e., a is orthogonal to b and/or b is orthogonal to c.

Overall, the approach (7) leads to a positive definite $\boldsymbol{M}$, and thus to a strictly negative Lyapunov function, if there exists an orthogonality condition between two successive tasks while an independency condition is sufficient for the remaining tasks. For the approach (9) the condition necessary and sufficient is that an independency condition (13) holds between the second and first tasks and between the third task and the augmented Jacobian obtained stacking the first two tasks.

A. Extension to N tasks

So far the 3-task case has been discussed. The generalization to a number N of tasks leads to the conclusion that, with the use of the equation eq. (9), the matrix $\boldsymbol{M}$ is always a lower block-triangular matrix. The sub-matrices on the main diagonal, however, are positive definite only if the condition (13) holds between each task and the Jacobian obtained by stacking all the higher-priority tasks. In the latter case, a choice of positive definite matrix gains leads to a negative definite Lyapunov function and thus to the convergence of all the task errors to zero.

When eq. (7) is used, however, the matrix $\boldsymbol{M}$ is not anymore guaranteed both to be a lower block-triangular matrix and to exhibit positive definite sub-matrices on the diagonal. In such a case the matrix $\boldsymbol{M}$ does not exhibit evident properties concerning its eventual positive/negative definiteness and general stability conclusions can not be made.

In conclusion, the approach (7) is stable when two tasks are considered and they are at least independent; in case of three tasks, it is required that there exists an orthogonality condition between two successive tasks, while an independency condition is sufficient for the remaining tasks. For more than three tasks no simple property exists. The approach (9), on the other side, can be used with the desired number of tasks as long as the independency condition (13) holds when an additional task is considered with respect to Jacobian obtained by stacking all the higher-priority tasks.

IV. Missions for Multi-robot systems

In the recent years, the NSB has been experimentally applied to a variety of missions involving a planar team of wheeled mobile robots [1], [2]. In this section, the designed missions will be recalled in terms of their corresponding task' functions [4] and priority orders, and the stability analysis will be discussed for the first time applying the results presented in this paper.

A. Spread Mission: Centroid + Variance

The first mission concerns the possibility to control a team of robots by assigning the position of its centroid and the variance around it. Assigning these two task variables, it is possible to influence the positioning and the spread of the platoon in the environment. In the following, the definitions of the task functions are reported and the stability of the mission composed by these two tasks is discussed.

1) Task function for platoon centroid: A task function for platoon centroid expresses the mean value of all the vehicles' positions as a synthetic data about the platoon location. In this case, the 2-dimensional task function $\boldsymbol{\sigma}_c$ for a team of n robots is simply given by:

$$\boldsymbol{\sigma}_c = \boldsymbol{f}_c(\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n) = \frac{1}{n}\sum_{i=1}^{n} \boldsymbol{p}_i = \overline{\boldsymbol{p}}. \qquad (18)$$

It is worth underlining that the subscripts of this and of the follower task functions is only connected to the function name and not to the priority order as in Section II.

The contribution of the vehicles' velocity to the task variation is

$$\dot{\boldsymbol{\sigma}}_c = \sum_{i=1}^{n} \frac{\partial \boldsymbol{f}_c(\boldsymbol{p})}{\partial \boldsymbol{p}_i}\boldsymbol{v}_i = \boldsymbol{J}_c(\boldsymbol{p})\boldsymbol{v}, \qquad (19)$$

where $\boldsymbol{v}_i \in \mathbb{R}^2$ is the velocity of the $i-$robot and the Jacobian matrix $\boldsymbol{J}_c \in \mathbb{R}^{2\times 2n}$ is

$$\boldsymbol{J}_c = \frac{1}{n}\begin{bmatrix} \cdots & 1 & 0 & \cdots \\ \cdots & 0 & 1 & \cdots \end{bmatrix} \qquad (20)$$

whose pseudoinverse is simply given by $\boldsymbol{J}_c^{\dagger} = n\boldsymbol{J}_c^{\mathrm{T}}$.

2) Task function for platoon variance: Together with the platoon centroid, it is of interest to consider the variance of all the vehicles' positions as a synthetic data on their spreading around the centroid.

The task function for platoon variance $\boldsymbol{\sigma}_v \in \mathbb{R}^2$ is defined as

$$\boldsymbol{\sigma}_v = \frac{1}{n}\sum_{i=1}^{n}\begin{bmatrix} (x_i - \bar{x})^2 \\ (y_i - \bar{y})^2 \end{bmatrix}, \qquad (21)$$

whose $(2 \times 2n)$ Jacobian is

$$\boldsymbol{J}_v = \frac{2}{n}\begin{bmatrix} \cdots & x_i - \bar{x} & 0 & \cdots \\ \cdots & 0 & y_i - \bar{y} & \cdots \end{bmatrix} \qquad (22)$$

and whose pseudoinverse is

$$\boldsymbol{J}_v^{\dagger} = \frac{n}{2}\begin{bmatrix} \vdots & & \vdots \\ \dfrac{x_i - \bar{x}}{\sum\limits_{j=1}^{n}(x_j - \bar{x})^2} & & 0 \\ 0 & & \dfrac{y_i - \bar{y}}{\sum\limits_{j=1}^{n}(y_j - \bar{y})^2} \\ \vdots & & \vdots \end{bmatrix}. \qquad (23)$$

The Jacobian $\boldsymbol{J}_v$ associated with the platoon variance may loose full rank at some system configuration, in this case, the use of some singularity-robust inverse kinematics, e.g., based on a damped least-squares inverse of the Jacobian (see [22]), would provide a smooth reference trajectory.

3) Stability: The spread mission is composed by the two previously defined tasks where the centroid has the highest priority. According to the results presented in this paper, it is possible to verify the orthogonality condition (12) between the two tasks:

$$\boldsymbol{J}_c\boldsymbol{J}_v^{\dagger} = \boldsymbol{O}_{2\times 2} \qquad (24)$$

where $\boldsymbol{O}_{2\times 2}$ is the (2×2) null matrix. Thus, for a two task mission, this condition is sufficient to prove that the matrix $\boldsymbol{M}$ of eq. (17) is positive definite and, thus, $\dot{V} < 0$. According to the Lyapunov stability theorem for autonomous systems [13], both the tasks' errors are thus asymptotically stable.

B. Formation Mission: Centroid + Rigid Formation

The second mission concerns the possibility to control a team of robots by assigning the position of its centroid and a rigid formation around it. The centroid task function is the same of Section IV-A, while the rigid formation task function is recalled in the following.

1) Task function for platoon rigid formation: The rigid formation task moves the vehicles to a predefined formation relative to the centroid. The task function is defined as:

$$\boldsymbol{\sigma}_f = \begin{bmatrix} \boldsymbol{p}_1 - \overline{\boldsymbol{p}} \\ \vdots \\ \boldsymbol{p}_n - \overline{\boldsymbol{p}} \end{bmatrix}, \qquad (25)$$

while the corresponding Jacobian matrix $\boldsymbol{J}_f \in \mathbb{R}^{2n\times 2n}$ is:

$$\boldsymbol{J}_f = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{O}_{2\times 2} \\ \boldsymbol{O}_{2\times 2} & \boldsymbol{A} \end{bmatrix}, \qquad (26)$$

where

$$\boldsymbol{A} = \begin{bmatrix} 1 - \frac{1}{n} & -\frac{1}{n} & \cdots & -\frac{1}{n} \\ -\frac{1}{n} & 1 - \frac{1}{n} & \cdots & -\frac{1}{n} \\ \vdots & \vdots & \ddots & \vdots \\ -\frac{1}{n} & -\frac{1}{n} & \cdots & 1 - \frac{1}{n} \end{bmatrix}. \qquad (27)$$

Since the Jacobian matrix is singular, the pseudoinverse can not be calculated as $\boldsymbol{J}^{\mathrm{T}}\left(\boldsymbol{J}\boldsymbol{J}^{\mathrm{T}}\right)^{-1}$ but as a matrix that verifies the following properties:

$$\boldsymbol{J}\boldsymbol{J}^{\dagger}\boldsymbol{J} = \boldsymbol{J}; \quad \boldsymbol{J}^{\dagger}\boldsymbol{J}\boldsymbol{J}^{\dagger} = \boldsymbol{J}^{\dagger}$$

and with $\boldsymbol{J}\boldsymbol{J}^{\dagger}$ and $\boldsymbol{J}^{\dagger}\boldsymbol{J}$ symmetric. Since $\boldsymbol{J}_f$ is symmetric and idempotent, $\boldsymbol{J}_f^{\dagger} = \boldsymbol{J}_f$.

The desired value $\boldsymbol{\sigma}_{f,d}$ of the task function describes the shape of the desired formation; that is, once defined the formation, the elements of $\boldsymbol{\sigma}_{f,d}$ represent the coordinates of each vehicle in the barycenter reference frame.

2) Stability: The stability for a mission composed by the centroid and rigid formation tasks is straightforward by observing that $\boldsymbol{J}_f \in \mathbb{R}^{2n \times 2n}$ is equal to the null space projector of the centroid task function :

$$\boldsymbol{J}_f = \boldsymbol{I} - \boldsymbol{J}_c^{\dagger}\boldsymbol{J}_c = \mathcal{N}(\boldsymbol{J}_c) \tag{28}$$

this implies, due to the matrix relationships above, that $\mathcal{R}(\boldsymbol{J}_f^{\dagger}) \subseteq \mathcal{N}(\boldsymbol{J}_c)$ without need for further computations.

It is worth noticing that the definition (25), also if easy to understand, is not well posed because the rows result linear dependent (thus the Jacobian has not full rank). To properly define the task function, one of the rows should be deleted. Considering this reduced task function and the relative Jacobian, it is easy verifying that $\boldsymbol{J}_c\boldsymbol{J}_f^{\dagger} = \boldsymbol{O}$ as for the previous mission, and, resorting again to the Lyapunov stability theorem for autonomous systems [13], it is possible to observe that both the tasks' errors are asymptotically stable.

C. Circular Mission: Centroid + Circular

The third mission concerns the possibility to pose the robots of the team on a circumference with fixed radius and center (without the need for specific positions along the circle). The mission is achieved trough the centroid task function, already defined, and the circular task function.

1) Task function for keeping a platoon on a circle: The n-dimensional task function

$$\boldsymbol{\sigma}_s = \begin{bmatrix} \dots & \frac{1}{2}(\boldsymbol{p}_i - \boldsymbol{c})^{\mathrm{T}}(\boldsymbol{p}_i - \boldsymbol{c}) & \dots \end{bmatrix}^{\mathrm{T}} \tag{29}$$

can be used to keep each vehicle of the platoon at a given distance r from a point $\boldsymbol{c} \in \mathbb{R}^2$ by setting

$$\boldsymbol{\sigma}_{s,d} = \begin{bmatrix} \dots & r^2/2 & \dots \end{bmatrix}^{\mathrm{T}}. \tag{30}$$

The corresponding Jacobian is given by the $(n \times 2n)$ matrix:

$$\boldsymbol{J}_s = \text{block diag}\{(\boldsymbol{p}_1 - \boldsymbol{c})^{\mathrm{T}}, \dots, (\boldsymbol{p}_n - \boldsymbol{c})^{\mathrm{T}}\} \tag{31}$$

whose pseudoinverse is the $(2n \times n)$ matrix:

$$\boldsymbol{J}_s^{\dagger} = \text{block diag}\left\{\frac{(\boldsymbol{p}_1 - \boldsymbol{c})}{(\boldsymbol{p}_1 - \boldsymbol{c})^{\mathrm{T}}(\boldsymbol{p}_1 - \boldsymbol{c})}, \dots, \frac{(\boldsymbol{p}_n - \boldsymbol{c})}{(\boldsymbol{p}_n - \boldsymbol{c})^{\mathrm{T}}(\boldsymbol{p}_n - \boldsymbol{c})}\right\}. \tag{32}$$

However, given the centroid as primary task, it is appropriate to modify the task function to keep the platoon on the circumference's circle in the 2-dimensional task function

$$\boldsymbol{\sigma}_s = \begin{bmatrix} \vdots \\ \frac{1}{2}(\boldsymbol{p}_i - \overline{\boldsymbol{p}})^{\mathrm{T}}(\boldsymbol{p}_i - \overline{\boldsymbol{p}}) \\ \vdots \end{bmatrix} \tag{33}$$

whose Jacobian can be computed to be

$$J_s = \begin{bmatrix} \left(1 - \frac{1}{n}\right)\left(\boldsymbol{p}_1 - \overline{\boldsymbol{p}}\right)^{\mathrm{T}} & -\frac{1}{n}\left(\boldsymbol{p}_1 - \overline{\boldsymbol{p}}\right)^{\mathrm{T}} & & -\frac{1}{n}\left(\boldsymbol{p}_1 - \overline{\boldsymbol{p}}\right)^{\mathrm{T}} \\ -\frac{1}{n}\left(\boldsymbol{p}_2 - \overline{\boldsymbol{p}}\right)^{\mathrm{T}} & \left(1 - \frac{1}{n}\right)\left(\boldsymbol{p}_2 - \overline{\boldsymbol{p}}\right)^{\mathrm{T}} & \dots & -\frac{1}{n}\left(\boldsymbol{p}_2 - \overline{\boldsymbol{p}}\right)^{\mathrm{T}} \\ & & \vdots & \\ -\frac{1}{n}\left(\boldsymbol{p}_n - \overline{\boldsymbol{p}}\right)^{\mathrm{T}} & & & \left(1 - \frac{1}{n}\right)\left(\boldsymbol{p}_n - \overline{\boldsymbol{p}}\right)^{\mathrm{T}} \end{bmatrix}$$

2) Stability: As for the spread mission, the stability of the circular mission is proved by verifying the orthogonality condition (12) between the two tasks:

$$\boldsymbol{J}_s\boldsymbol{J}_c^{\dagger} = \boldsymbol{O}_{2 \times 2}. \tag{34}$$

and resorting again to Lyapunov stability theorem for autonomous systems [13].

D. Escorting Missions: Centroid + Circular + Perimeter

In an escorting mission the platoon might be asked to keep the target at the centroid while the single vehicles lay at the same distance from the target and minimize the intruding possibilities of an external agent. Dually, the same requirement concerns an entrapment mission.

With reference to the planar case, this requirement is satisfied by placing the n vehicles at the vertices of a regular polygon of order n, the sides of which, thus, define a sort of intrusion/escape window. A possible way to achieve this formation is to keep the vehicles on a suitable circle through the above-defined task functions for the circular mission, while maximizing the mutual distance between adjacent vehicles. The latter task can be achieved by properly assigning the perimeter of the polygon inscribed in the circle; in fact, a regular polygon has the maximum perimeter among all the polygons of the same order inscribed in a given circle.

1) Task function for perimeter: Let us define a n-dimensional vector $\boldsymbol{k}$ that collects the indexes to the vehicles in their order along the circle. Thus, k_j (i.e., the j-th coordinate of vector $\boldsymbol{k}$) is the index that identifies the vehicle at the j-th place along the circle —not necessarily the j-th vehicle— and k_j, k_{j+1} index two consecutive vehicles along the circle. Obviously, any vehicle can be chosen as k_1 and k_n, k_1 are consecutive.

The sought task function is thus given by a measurement of the perimeter, such as

$$\sigma_p = \frac{1}{2}(\boldsymbol{p}_{k_1} - \boldsymbol{p}_{k_n})^{\mathrm{T}}(\boldsymbol{p}_{k_1} - \boldsymbol{p}_{k_n}) + \\ + \frac{1}{2}\sum_{j=2}^{n}(\boldsymbol{p}_{k_j} - \boldsymbol{p}_{k_{j-1}})^{\mathrm{T}}(\boldsymbol{p}_{k_j} - \boldsymbol{p}_{k_{j-1}}), \tag{35}$$

whose Jacobian $\boldsymbol{J}_p \in \mathbb{R}^{1 \times 2n}$ is

$$\boldsymbol{J}_p = \begin{bmatrix} \dots & \left((\boldsymbol{p}_{k_j} - \boldsymbol{p}_{k_{j-1}})^{\mathrm{T}} + (\boldsymbol{p}_{k_j} - \boldsymbol{p}_{k_{j+1}})^{\mathrm{T}}\right) & \dots \end{bmatrix}. \tag{36}$$

It is worth noticing that the vector $\boldsymbol{k}$ dynamically changes during the mission, i.e., no predefined position is imposed on the vehicles.

At this point, for a platoon of n vehicles that must entrap a target while guaranteeing a given length l of the escape window, the radius of the circle in (30) must be set to

$$r = \frac{l}{2\cos\left(\frac{\pi}{2} - \frac{\pi}{n}\right)} \qquad (37)$$

and the desired value of the perimeter is simply $\sigma_{p,d} = n\,l$.

On the other hand, for a platoon of n vehicles that must entrap a target while guaranteeing a given (safety) distance r of the vehicles from the target, the radius of the circle in (30) is simply equal to r itself and the desired value of the perimeter is

$$\sigma_{p,d} = 2\,n\,r\cos\left(\frac{\pi}{2} - \frac{\pi}{n}\right)\,. \qquad (38)$$

2) Stability: The escorting mission is realized with the previous three tasks in following priority order:

1) Centroid;   2) Circular;   3) Perimeter.

According to the results presented in this paper, the stability of a three task mission is proved showing that the diagonal blocks of the matrix $\boldsymbol{M}$ of eq. (17) are positive definite and the sub-matrix $\boldsymbol{M}_{23}$ is null. Resorting to symbolic or numerical instruments, it is possible to analytically verifying that the three tasks of the escorting mission are independent and that the first two tasks are orthogonal as long as initial configuration is not a singular configuration (e.g., all the vehicles are along a line) and $n > 2$. Following the considerations of Section III, these conditions are sufficient to prove the simultaneously asymptotic stability of the three tasks' errors. Moreover, analogous deductions can be done inverting the priorities of Centroid and Circular task functions.

V. Conclusions

The use of behavior-based approaches are of great importance in mobile robotics applications because they give the system the possibility to navigate in complex environments and achieve widely different kind of missions. Despite the advantage related to the flexibility of behavior-based approaches, for most of them a rigorous and general stability analysis, that ensures the possibility to effectively achieve the defined behaviors, is missing. A new behavior-based approach, namely the Null-Space based Behavioral control, has been recently proposed and tested with different robotic systems. This approach results flexible to achieve widely different kind of missions with different robotic systems and it has been extensively tested with a multi-robot experimental set-up. In this paper, a rigorous stability analysis, proving what kind of missions can be fulfilled, and a general procedure to verify stability conditions have been presented in order to underline the mathematical consistency of the approach. Moreover, the stability of several missions for multi-robot systems has been discussed.

References

[1] G. Antonelli, F. Arrichiello, S. Chakraborti, and S. Chiaverini. Experiences of formation control of multi-robot systems with the Null-Space-based Behavioral control. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 1068–1073, Rome,I, Apr. 2007.

[2] G. Antonelli, F. Arrichiello, and S. Chiaverini. An experimental study of the entrapment/escorting mission for a multi-robot system. *IEEE Robotics and Automation Magazine (RAM). Special Issues on Design, Control, and Applications of Real-World Multi-Robot Systems*, 15(1):22–29, March 2008.

[3] G. Antonelli, F. Arrichiello, and S. Chiaverini. The Null-Space-based Behavioral control for autonomous robotic systems. *Journal of Intelligent Service Robotics*, 1(1):27–39, online March 2007,printed January 2008.

[4] G. Antonelli and S. Chiaverini. Kinematic control of platoons of autonomous vehicles. *IEEE Transactions on Robotics*, 22(6):1285–1292, Dec. 2006.

[5] R.C. Arkin. Motor schema based mobile robot navigation. *The International Journal of Robotics Research*, 8(4):92–112, 1989.

[6] R.C. Arkin. *Behavior-Based Robotics*. The MIT Press, Cambridge, MA, 1998.

[7] P. Baerlocher and R. Boulic. Task-priority formulations for the kinematic control of highly redundant articulated structures. In *1998 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1998.

[8] T. Balch and R.C. Arkin. Behavior-based formation control for multirobot teams. *IEEE Transactions on Robotics and Automation*, 14(6):926–939, 1998.

[9] C. Belta and V.K. Kumar. Abstraction and control of groups of robots. *IEEE Transactions on Robotics*, 20(5):865–875, 2004.

[10] R.A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of Robotics and Automation*, 2:14–23, 1986.

[11] S. Chiaverini. Singularity-robust task-priority redundancy resolution for real-time kinematic control of robot manipulators. *IEEE Transactions on Robotics and Automation*, 13(3):398–410, 1997.

[12] G.H. Golub and C.F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, Baltimore, MD, third edition, 1996.

[13] H.K. Khalil. *Nonlinear Systems*. Prentice-Hall, Upper Saddle River, New Jersey, 2nd edition, 1996.

[14] A. Liégeois. Automatic supervisory control of the configuration and behavior of muldibody mechanisms. *IEEE Transactions on Systems, Man and Cybernetics*, 7:868–871, 1977.

[15] A.D. Mali. On the behavior-based architectures of autonomous agency. *IEEE Transactions on Systems, Man and Cybernetics*, 32(3):231–242, Aug. 2002.

[16] N. Mansard and F. Chaumette. Task sequencing for high-level sensor-based control. *IEEE Transactions on Robotics and Automation*, 23(1):60–72, 2007.

[17] M.J. Mataric. Behavior-based control: Examples from navigation, learning, and group behavior. *Journal of Experimental and Theoretical Artificial Intelligence*, 9(2-3):323–336, 1997.

[18] L.E. Parker. On the design of behavior-based multi-robot teams. *Advanced Robotics*, 10(6):547–578, 1996.

[19] P.E. Rybski, N.P. Papanikolopoulos, S.A. Stoter, D.G. Krantz, K.B. Yein, M. Gini, R. Voyles, D.F. Hougen, B. Nelson, and M.D. Ericksn. Enlisting rangers and scouts for reconnaissance and surveillance. *IEEE Robotics & Automation Magazine*, 7(4):14–24, 2000.

[20] B. Siciliano. Kinematic control of redundant robot manipulators: A tutorial. *Journal of Intelligent Robotic Systems*, 3:201–212, 1990.

[21] H.G. Tanner, G.J. Pappas, and V. Kumar. Leader-to-formation stability. *IEEE Transactions on Robotics and Automation*, 20(3):443–455, 2004.

[22] C.W. Wampler II. Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods. *IEEE Transactions on Systems, Man and Cybernetics*, 16(1):93–101, 1986.

[23] M.M. Zavlanos and G.J. Pappas. Dynamic Assignment in Distributed Motion Planning With Local Coordination. *IEEE Transactions on Robotics*, 24(1):232–242, 2008.