

A Model Based Approach to Modular Multi-Objective Robot Control

Petter Ögren · John W. C. Robinson

Received: 13 August 2010 / Accepted: 15 December 2010 / Published online: 18 January 2011
© The Author(s) 2011. This article is published with open access at Springerlink.com

Abstract Two broad classes of robot controllers are the modular, and the model based approaches. The modular approaches include the Reactive or Behavior Based designs. They do not rely on mathematical system models, but are easy to design, modify and extend. In the model based approaches, a model is used to design a single controller with verifiable system properties. The resulting designs are however often hard to extend, without jeopardizing the previously proven properties. This paper describes an attempt to narrow the gap between the flexibility of the modular approaches, and the predictability of the model based approaches, by proposing a modular design that does the combination, or arbitration, of the different modules in a model based way. By taking the (model based) time derivatives of scalar, Lyapunov-like, objective functions into account, the arbitration module can keep track of the time evolution of the objectives. This enables it to handle objective tradeoffs in a predictable way by finding controls that preserve an important objective that is currently met, while striving to satisfy another, less important one that is not yet achieved. To illustrate the approach a UAV control problem from the literature is solved, resulting in comparable, or better, performance.

Keywords Control · Robotics · Reactive control · Behavior-based systems · Lyapunov function · UAV

Mathematics Subject Classifications (2010) 70E60 · 37B25 · 93C23

P. Ögren (✉) · J. W. C. Robinson
Department of Autonomous Systems, Swedish Defence Research Agency (FOI), 16490
Stockholm, Sweden
e-mail: petter.ogren@foi.se, petter.ogren@gmail.com

1 Introduction

The field of robotic control can loosely be divided into *model based* and *modular* control approaches. In model based control, a mathematical model of the dynamic or kinematic behavior of the robot is used to synthesize a single controller and analyze and prove attractive system properties such as safety or convergence to a goal location. Model-based approaches to robot control with multiple objectives include [13, 14, 21, 24]. These approaches perform well, but are tailor made to the specific problems they address, and adding additional objectives to be met is often difficult or impossible without spoiling the structures being used in the proofs.

The modular control methods include [3, 4, 7–11, 15, 17–19, 22, 23, 25], and a nice overview can be found in [1]. These methods are perhaps more often referred to as *Behavior Based*, or *Reactive*, but throughout this paper we will call them *modular* in order to emphasize this key property. In modular approaches, each controller is designed with a specific control objective in mind, more or less independently from the rest of the controllers. The outputs of the controllers are then merged in some clever fashion in an arbitration module that outputs the overall control action, as illustrated in Fig. 1.

This merging, or arbitration, can be either a selection of the most important controller at each time instant, or a fusion of a set of control signals into a single output. Examples of selection schemes are [4, 5, 10], while fusion methods include [7, 8, 15, 17, 19]. The modular methods have received a lot of attention due to the fact that they are flexible, robust and easy to use. However, it is well known [23], that it often gets hard to predict the overall system behavior when more modules are added.

In this paper, which is an extension of our work in [12], we attempt to narrow the gap between the modular approaches and the model based methods. The proposed controller will thus retain the modular structure of Fig. 1, but do the arbitration in a more predictable and model based fashion, inspired by Control Lyapunov Functions (CLFs) [2]. This is done by letting each individual *Control Module Output*, see Fig. 1, include not only its preferred control signal, but also a scalar function measuring to what extent the objective of that module is met. The arbitration then keeps track of, and works towards satisfying, these objectives in a predictable way. This is done by making sure that the (model based) time derivatives of the scalar functions satisfy certain inequalities, reflecting the user priorities regarding the corresponding objectives.

In order to formalize the user priorities, we first note that all modular approaches address a more or less implicit *multi objective control problem*. We then formally state this problem, and include a scalar function and a user defined order of importance for each control objective. As indicated above, the scalar functions and user priorities are then used in the model based arbitration.

We illustrate the approach by solving an example problem taken from [21], where a solution to a multi objective UAV surveillance problem is presented. Applying the proposed approach to the problem we get controllers that are modular and flexible, but also somewhat less conservative than previous solutions, due to the fact that the bounds are handled explicitly.

As noted above, this work is an extension of [12]. In this paper, the theoretical material has been thoroughly revised, including definitions, lemmas and proposition,

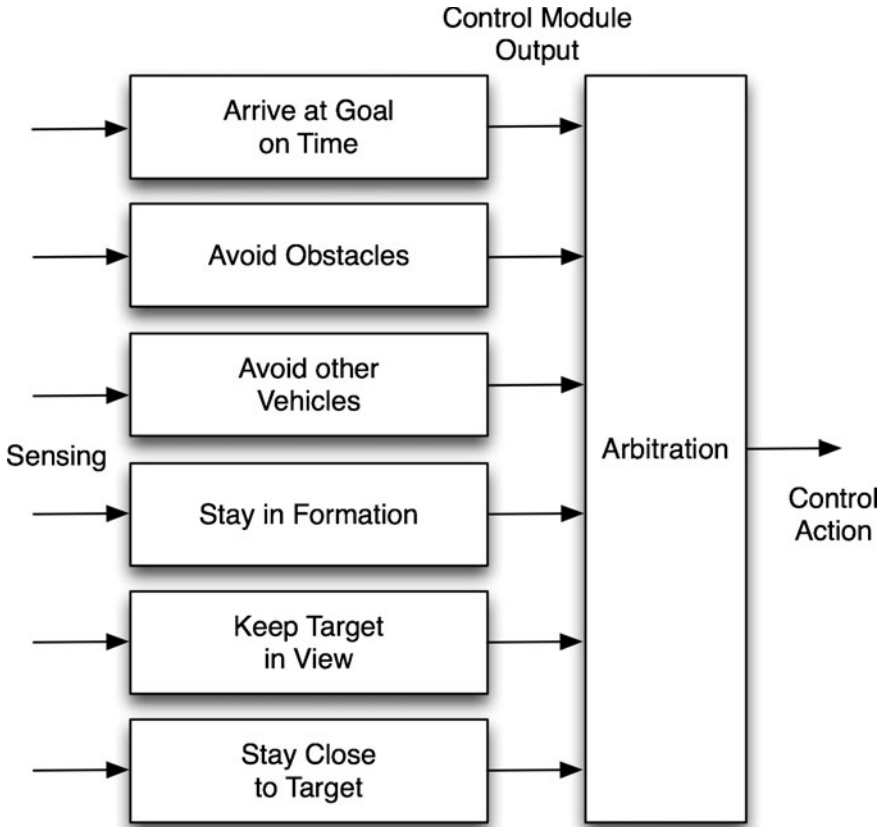


Fig. 1 A typical modular control architecture, as described in [1]. The format of the *Control Module Output* depends on the type of Arbitration being used, as explained in Section 2 below. The proposed approach aims at making this arbitration in a more model based fashion

and the simulation section is completely new, with a complex UAV problem, taken from [21], replacing the ground robot obstacle avoidance problem investigated in [12].

The organization of this paper is as follows. In Section 2 we review the structure of some existing modular control architectures. Section 3 then formalizes the Multi Objective Control Problem and Section 4 presents the proposed modular model based controller design. Finally, the approach is illustrated by a UAV example in Section 5 and conclusions are drawn in Section 6.

2 Background: Modular Control Architectures

In this Section, we will describe a number of modular control approaches that will be used as inspiration in our attempt to find a *model based* modular architecture. As noted above, the modular approaches are sometimes also called Reactive, or

Behavior Based, but in this context their key property is modularity, hence the name *modular* for them used in this paper.

A typical modular architecture is seen in Fig. 1. The different controllers work in parallel, supplying the *Arbitrator* with *Control Module Outputs*. These are then merged in some fashion to produce a control action. The key property of the modular approaches is not that their *execution* is modular, but that they facilitate modular *design*. This means that each module can be designed without knowledge of the other modules, and that the arbitration can be designed without knowledge of what individual control modules it will be connected to. The only part of the design process that takes the whole setup into account is when the user sets a number of parameters in the arbitration module, typically describing some form of user priorities.

Note that modular *design* is a very attractive and important feature. This is due to the fact that the design of robot systems is often an evolutionary process and the robot itself, as well as its tasks and its environment often change over time [23]. For example, new sensors might be added to the robot, new tasks might be added to a typical mission, and the environment in which the missions are performed might change. In such circumstances, a modular design enables not only the addition of functionality, but also the possibility of switching control modules on and off depending on the situation. Finally, the importance of modular design can be deduced from the apparent popularity of all the modular approaches [3, 4, 7–11, 15, 17–19, 22, 23, 25].

We will now give a brief description of five important arbitration schemes, from which the proposed design borrows ideas and inspiration.

1. In *Suppression* [5], the *Control Module Output* of Fig. 1 is a suggested control action and a binary value indicating whether or not the module is applicable to the situation. The *Arbitration* then reviews a given priority list and picks the behavior with highest priority of the active ones, and executes the corresponding control action. For example, *Avoid-obstacle* can override *Arrive-at-Goal-on-Time* if there is an obstacle close by.
2. In *Selection* [10], the *Control Module Output* of Fig. 1 is a suggested control action and a scalar *activation level* indicating how urgent the proposed action is. The *Arbitration* then picks the behavior with highest activation level, and executes the corresponding control action.
3. In *Voting* [18], the *Behavior Output* is a list where a number of votes has been distributed over a finite set of control choices. The *Arbitration* then sums up all votes from the different behaviors and executes the choice with the most votes.
4. In *Vector summation* [7], the *Behavior Output* is in the form of a desired control action, usually a vector corresponding to the desired velocity or acceleration. The *Arbitration* then sums all vectors to get the executed control action.
5. In *Spikes and Zones* [15], the *Behavior Output* includes both *spikes*, desired control outputs, and *zones*, acceptable intervals for the control output. The *Arbitration* then tries to find a spike inside the intersection of all zones. If there is no such spike it chooses the control signal closest to the current one inside the intersection. If the intersection is empty the zones of lower priority are ignored until the intersection is non-empty.

In the proposed approach we will use inspiration from all of the above. The priority list of *Suppression* is turned into a priority table, such as Table 1, where not

Table 1 A priority table containing the values of b_{ij}

b_{ij}	0	1	2	3	4
$V_1 \leq$	∞	a	a	c	$-\infty$
$V_2 \leq$	∞	∞	b	b	$-\infty$

Each row corresponds to a scalar objective function V_i , while each column corresponds to a different priority level. The problem corresponding to this Table is illustrated in Fig. 2

only objectives, but also different levels of objective satisfaction is given different priorities. The notion of an *active* behavior is translated into checking whether a high priority objective is close to being violated or not. In *Selection*, the activation levels depend on how close an objective is to being violated. We incorporate such ideas by measuring how close an objective is to being violated and adjusting the control constraints accordingly. The idea behind *Voting* and *Spikes and Zones*, that a behavior can transmit information on what actions are best, but also acceptable relative to the behavior goal, is incorporated into the constraints on the time derivatives of the scalar functions measuring objective satisfaction. The brief statements above will be discussed in more detail once we have presented the proposed approach. However, first we state the problem being addressed.

3 A Multi Objective Control Problem

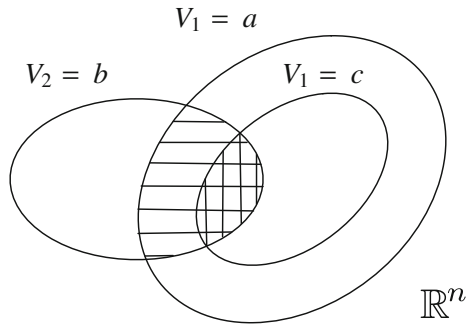
In this section we will formalize what we mean by a multi objective control problem. As hinted above, there is no point in using a modular approach of the type illustrated in Fig. 1 if the underlying control problem does not include multiple objectives. Examples of such objectives can be found in Fig. 1 and include *Arrive at Goal on Time*, *Avoid Obstacles* and *Stay in formation*. In some situations, these objectives will be contradictory, e.g., if two vehicles have different goal locations, *Arrive at Goal on Time* and *Stay in formation* can probably not be simultaneously satisfied throughout the execution. Thus, an important task for the arbitration is to handle such tradeoffs. Therefore, the problem formulation must include user priorities, based on which the tradeoffs can be made.

To formalize objectives and priorities, we assign a scalar objective function V_i and a set of bounds b_{ij} of increasing difficulty to each objective. The bounds b_{ij} can be listed in a *priority table*, such as Table 1, where the first priority objectives are found in the first column and so on. The columns thus correspond to different intersections of level sets of the V_i s, as illustrated in Fig. 2. Formally we write as follows.

Problem 1 (A Multi Objective Control Problem) Suppose we have a control affine dynamic system $\dot{x} = f(x) + g(x)u$, with $x \in \mathbb{R}^n$, $u \in U_{\text{adm}} \subset \mathbb{R}^m$ where U_{adm} is a closed, bounded and convex set of admissible controls, and a set of piecewise C^1 functions $V_i : \mathbb{R}^n \rightarrow \mathbb{R}$, and constants $b_{ij} \in \mathbb{R}$ where $b_{ij} \geq b_{i(j+1)}$ $i \in \{1 \dots N_i\}$, $j \in \{0 \dots N_j\}$. Then, let the *Current Level of Satisfaction* be

$$\bar{j}(x) = \max\{j : V_i(x) \leq b_{ij}, \forall i\}, \tag{1}$$

Fig. 2 An example multi objective control problem, corresponding to the priorities in Table 1. The *horizontal stripes* correspond to column 2 and the *vertical stripes* to column 3. Thus, the control objective in the example is to bring the state of the system from some x_0 , to satisfy the bounds b_{ij} in the order of Table 1, and finally reaching the vertically striped area



and the *Multi Objective Control Problem* at time t_0 is to solve the following optimization

$$\begin{aligned}
 & \min_u T, \\
 & \text{s.t. } \bar{j}(x(T)) > \bar{j}(x(t_0)), \\
 & \quad \bar{j}(x(t)) \geq \bar{j}(x(t_0)), t \in [t_0, T].
 \end{aligned} \tag{2}$$

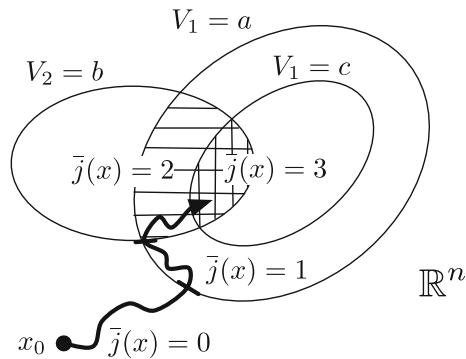
□

Thus, the current level of satisfaction $\bar{j}(x)$ is the rightmost column of the priority table where all bounds are satisfied at state x , and the Multi Objective Control Problem corresponds to increasing $\bar{j}(x)$ in minimum time T , without allowing it to decrease, even momentarily, on the way. We conclude this section with an example execution of a multi objective control problem.

Example 1 Suppose we have two scalar functions V_1, V_2 measuring objective satisfaction and the set of bounds b_{ij} given in Table 1. The two *rows* of the table correspond to V_1 and V_2 and the *columns* of the table correspond to different priority levels. Looking at the table we see that the zero:th priority (which is always satisfied) in column 0 is to have $V_1 \leq \infty, V_2 \leq \infty$. Then, the first priority, in column 1, is to achieve $V_1 \leq a$. Similarly, the second priority, in column 2, is to achieve $V_2 \leq b$, while keeping $V_1 \leq a$. Finally, the third priority, in column 3, is to achieve $V_1 \leq c \leq a$, while keeping $V_2 \leq b$. Consider Fig. 3, and suppose the system starts in some x_0 , with $\bar{j}(x_0) = 0$. Now imagine that some version of the arbitration method *Selection* was used. Then, we would first apply some controller tailored to the first objective, until $V_1 \leq a$, making $\bar{j}(x) = 1$. Then, some other controller might be selected until $V_2 \leq b$, and $\bar{j}(x) = 2$. Finally, the first controller might be applied again until $V_1 \leq c$ and $\bar{j}(x) = 3$.

Remark 1 Note that, when $\bar{j}(x) = 1$ in Fig. 3, the bound $V_1 \leq a$ is almost violated. This would have reduced $\bar{j}(x)$ and caused a possible chattering between the two

Fig. 3 An illustration of Example 1. A series of Multi Objective Control Problems are solved until $\bar{j}(x) = 3$. A simulation plot corresponding to this illustration can be found in Fig. 6



controllers. To avoid such effects, the modular approach proposed below takes the time derivative \dot{V}_i into account, as well as the size of the available slack, $b_{ij} - V_i$, when addressing the multi objective control problem.

Remark 2 Note that a number of *model based* approaches could be applied to the problem defined above. However, in this paper we are looking for a *modular* model based controller, of the type illustrated in Fig. 1. The advantage of a *modular* design is, as explained in Section 2, that it can be extended if Problem 1 is extended with an additional objective.

Remark 3 Finally, note that it is quite straightforward to create a Multi Objective Control Problem that is impossible to solve. For instance, for any V_1 , if $V_2 = -V_1$ and $b_{11} = b_{12} = -1$, then we can not have $V_1 \leq -1$ and $-V_1 \leq -1$ at the same time and thus $\bar{j}(x) = 0$ for all x . Thus, it is pointless to look for an arbitration scheme that solves all instances of Problem 1.

Having made these remarks, we are ready to look at the proposed design.

4 Proposed Modular Solution

In this section we will describe the proposed arbitration module and state two propositions showing that it is a reasonable choice for doing modular multi objective control. Before going into details, we give an overview of the ideas and results.

Informally, with a slight abuse of notation, the basic arbitration idea is as follows. First, given that we are at state x the current level of satisfaction $\bar{j}(x)$ is determined, i.e., the appropriate column of the priority table is located. For technical reasons explained in Remark 4 below, we then re-label $\bar{j}(x)$ to $j^*(x)$. Then, the set of acceptable control choices u are narrowed down to U^* , the controls satisfying $\dot{V}_i(x, u) \leq k(b_{ij^*(x)} - V_i(x))$, $k > 0$, i.e., the increase in V_i is bounded by the size of the slack $b_{ij^*} - V_i$. Then, we analyze the priority table to find $I^*(x)$, the objective(s)

to be focused on next. Finally, a control is chosen inside U^* that is close to $u_i, i \in I^*$, the controls that were tailored to the objective(s) in I^* .

The definitions below will be followed by a set of performance results. We can not aim for an arbitration design that solves the general problem, since it was noted in Remark 3 that by assigning contradicting objectives one could easily create instances of Problem 1 that can not be solved. Instead, we will show the following results: First, that the proposed arbitration scheme only violates a satisfied objective when necessary, due to contradicting objectives (Proposition 1). Second, that it does indeed solve Problem 1 and progress through the priority table if no objective tradeoffs occur (Proposition 2).

4.1 Proposed Arbitration Module

In the proposed design, we will keep the modular structure shown in Fig. 1, but let the *Control Module Output* include not only a control signal u_i that is tailored to objective i , but also the scalar functions $V_i(x)$. Formally, we define the controller as follows.

Definition 1 Given a Multi Objective Control Problem, and a set of different controllers u_i and scalars $k > 0$, let

$$u^* = \operatorname{argmin}_{u \in U^*} \sum_{i \in I^*} \|u - u_i\|^2 \tag{3}$$

where

$$U(x, j) = \{u \in U_{\text{adm}} : \dot{V}_i(x, u) \leq k(b_{ij} - V_i(x)), \forall i\} \tag{4}$$

$$j^*(x) = \max\{j : V_i(x) \leq b_{ij}, \forall i, U(x, j) \neq \emptyset\} \tag{5}$$

$$U^*(x) = U(x, j^*) \tag{6}$$

$$I^*(x) = \{i : b_{ij^*(x)} > b_{i(j^*(x)+1)}\}. \tag{7}$$

Informally, Expression 3 corresponds to choosing a control that is close to $u_i, i \in I^*$, in order to increase $\bar{j}(x)$, which at the same time is inside U^* , to make sure that $\bar{j}(x)$ does not decrease. The rationale behind the choice of $U(x, j)$ in Expression 4 can be seen from Proposition 1 below, but the fact that it gives upper bounds on \dot{V}_i in terms of the margins $b_{ij} - V_i$ indicates that it is related to keeping $\bar{j}(x)$ from decreasing. Finally, Expression 7 selects the objective(s) that is focused on next, to increase $\bar{j}(x)$.

Remark 4 The added condition on $j^*(x)$ in Expression 5, compared to $\bar{j}(x)$ in Eq. 1, is introduced to make sure that $U^*(x) \neq \emptyset$ which makes u^* well defined (see Lemma 1 below).

Remark 5 Note that the control u^* above is found by solving a pointwise optimization problem. This can often be done very efficiently using optimization software since $u \in U_{\text{adm}} \subset R^m$ in Problem 1 is often of low dimensionality and the objective function is quadratic, see Remark 7.

4.2 Properties of the Proposed Controller

We will now investigate the properties of the proposed controller in a set of propositions and lemmas. Proposition 1 will show that $\bar{j}(x)$ only decreases when necessary, and Proposition 2 will show that in the ideal case, when no tradeoffs have to be made between the different objectives, all objectives will be achieved in finite time. Note that since Definition 1 is quite general, see Remark 3, it is hard to prove properties that are stronger than the ones above. Note also, that these two properties roughly correspond to stability and asymptotic stability for the Multi Objective Control Problem.

Apart from the two propositions, we will also state two lemmas. The first shows that the proposed controller is well defined and the second is related to the prerequisites of Proposition 2.

Lemma 1 *The control u^* in Definition 1 is well defined.*

Proof We first note that the maximization in j^* is always well defined since $U_{\text{adm}} \neq \emptyset$ and $b_{i0} = \infty$. Next, since $\dot{V}_i(x, u) = (\partial V_i(x)/\partial x)(f(x) + g(x)u)$ and U_{adm} is closed, bounded and convex, so is U^* , and this makes the minimization in Eq. 3 well defined. Finally, the fact that $b_{iN_j} = -\infty$ means that $j^* < N_j$ which in turn makes I^* well defined. □

To establish the conditions for nondecreasing $\bar{j}(x)$ and finite time maximization of satisfaction, as discussed above, we need the following definition.

Definition 2 (Non-arbitrable States) All states x such that

$$j^*(x) \neq \bar{j}(x)$$

are called *non-arbitrable states*.

Note that this happens when $U(x, \bar{j})$ is empty, i.e., there is no $u \in U_{\text{adm}}$ such that $\dot{V}_i(x, u) \leq k(b_{i\bar{j}} - V_i(x))$ for all i , meaning that the state x is not arbitrable.

Proposition 1 *Suppose that there are no non-arbitrable states in the system, i.e.,*

$$j^*(x) = \bar{j}(x), \quad \forall x. \tag{8}$$

Then, when using the control u^ none of the currently achieved objectives will be violated in the future, i.e. if $x(t)$ denotes the corresponding closed loop solution then*

$$j^*(x(t_0)) \leq j^*(x(t)) \tag{9}$$

for all $t \geq t_0$.

Proof Suppose that Eq. 8 holds. Then we have

$$j^*(x(t)) = \bar{j}(x(t)) = \max\{j : V_i(x(t)) \leq b_{ij}, \forall i\}, \quad t \geq t_0,$$

and we define $j_0^* = j^*(x(t_0))$.

Assume $j^*(x(t)) < j_0^*$ for some $t > t_0$ then $\bar{j}(x(t)) = j^*(x(t)) < j_0^*$ and thus $V_i(x(t)) > b_{ij_0^*}$ for some i . Define

$$t' = \inf\{t \geq t_0 : \exists i : V_i(x(t)) > b_{ij_0^*}\}. \tag{10}$$

Then, by continuity of the involved functions there exists an index i' such that

$$\dot{V}_{i'}(x(t'), u^*(t')) > 0, \tag{11}$$

and at the same time

$$V_i(x(t')) \leq b_{ij_0^*}, \forall i. \tag{12}$$

However, Eq. 12 implies $j^*(x(t')) = j_0^*$ and $\dot{V}_{i'}(x(t'), u^*(t')) \leq k(b_{i'j_0^*} - V_{i'}(x(t'))) = 0$, so we have a contradiction. \square

Having investigated under what conditions the already satisfied objectives are preserved, we now analyze when the rest of the objectives are achieved. Note however that, as suggested above, Definition 1 is very general and that the ideal setting below will not occur in most applications. Thus, the Proposition is primarily a sanity check of the proposed controller.

Proposition 2 *Suppose that for an isolated given i , the bounds $V_i \leq b_{ij}$, $j = 1 \dots N_j$ are all achieved in finite time if the control u_i is applied. Suppose further that, throughout the execution no non-arbitrable states are encountered, i.e. we have $j^*(x) = \bar{j}(x)$, that there is a always unique next objective $|I^*| = 1$, and $u_i \in U^*$ for $i \in I^*$, i.e., the corresponding control does not violate the previously achieved objectives.*

Then, all the objectives listed in the b_{ij} -table, except the last column, will be met in finite time, i.e. there exists a $t \geq t_0$ such that

$$V_i(x(t)) \leq b_{ij}, \quad \forall i, j < N_j.$$

Proof From Eq. 3 and the fact that $u_i \in U^*$ we have that the implemented control u^* will be equal to the different u_i , which in turn are known to satisfy their objectives in finite time. Furthermore, since $u^* \in U^*$ we know from Proposition 1 that none of the achieved objectives will be violated. Finally, since there are a finite number of columns of the table, all objectives will be satisfied in finite time. \square

Lemma 2 *Suppose that the system with the state feedback control $u = u_i$ has been shown to be asymptotically stable at the equilibrium point x^* using the CLF V_i with $V_i(x^*) = a < b_i$. Then the finite time property of Proposition 1 holds.*

Proof By the definition of asymptotic stability, we know that for each bound $b_i > a$, there is a finite time t' such that the solution $x(t)$ for the closed loop system satisfies $V_i(x(t)) < b_i$ for $t > t'$. \square

Finally, we make a note on the use of the only design parameter present in proposed approach, i.e., k .

Remark 6 The parameter k of Eq. 4 governs how fast each V_i can approach a bound b_{ij} . Clearly, if we have equality in Eq. 4, V_i would approach b_{ij} exponentially, with

the time constant $1/k$. Note also that it would be perfectly reasonable to replace k with a set of parameters k_i , one for each V_i , if such fine tuning is desired, perhaps to avoid having to re-scale the V_i s. An example of the exponential convergence can be seen in Fig. 11, and is discussed in Remark 8.

4.3 Relation to Previous Modular Control Approaches

In the next section we will see how the proposed controller performs when applied to a multi objective UAV surveillance problem. However, we will first comment on how this control structure compares to some other arbitration alternatives, including those that were reviewed in Section 2. The priority table clearly borrows a lot from the arbitration methods *Selection* and *Suppression*, both addressing the question of what to focus on "right now". Using the set U^* is similar to the ideas behind *Voting* and *Spikes and Zones*, in that options that are acceptable relative to many mission objectives are favored. Furthermore, minimizing a sum of $\|u - u_i\|^2$ resembles the arbitration method *Vector summation* in that it is a sum, and it resembles *Spikes and Zones* in that it tries to stay close to control signals preferred by individual control modules. Finally, there is a typographical resemblance between the proposed controller and the *Dynamic Systems Approach* [8], due to the fact that the latter is the most mathematical of the previous modular approaches. In the [Appendix](#) we analyze this resemblance in more detail.

Thus, the proposed controller is *modular*, but *model based* in the sense that it does the arbitration in a model based way inspired by CLFs. The structure of this arbitration is however, as described above, inspired by a set of different ideas from modular robot control.

5 A Surveillance UAV Example

In [12], a standard ground robot navigation example was presented, but here we apply the proposed approach to a more challenging example, taken from the literature, [21]. Here, a small Unmanned Aerial Vehicle (UAV) is to track a target on the ground using a fixed camera mounted in such a way that it is pointing down the wing. To make the problem more interesting the strength of the wind is smaller than, but on the same order of magnitude as the UAV air speed.

Since target motion enters the equations in the same way as wind, we let the wind term account for both effects. Following [21], the equations of motion of the UAV are

$$\dot{z}_x = V_a \cos \psi + V_w \cos \chi \tag{13}$$

$$\dot{z}_y = V_a \sin \psi + V_w \sin \chi \tag{14}$$

$$\dot{\psi} = \frac{g}{V_a} \tan \phi \tag{15}$$

$$\dot{\phi} = \alpha_\phi (\phi^c - \phi) \tag{16}$$

where (z_x, z_y) is the position, ψ is heading, χ is wind direction, V_a is airspeed, V_w is wind speed, g is the gravitational constant, ϕ is roll, α_ϕ is the auto pilot time constant and ϕ^c is the commanded roll. Note that airspeed, as well as wind direction

and magnitude are taken to be constant. We now make the following simplification. Assuming that the controlled roll angle dynamics is significantly faster than the heading dynamics we think of $\dot{\psi}$ as the control u , and let

$$\dot{\psi} = u, \quad |u| \leq \frac{g}{V_a} \tan \bar{\phi}. \tag{17}$$

where $\bar{\phi}$ is the maximal roll angle.

Using polar coordinates ρ, η to express the target position in the UAV frame, see Fig. 4, we get

$$\dot{\rho} = -V_a \sin(\eta) - V_w \sin(\eta - \chi + \psi) \tag{18}$$

$$\dot{\eta} = -u - \frac{V_a}{\rho} \cos(\eta) - \frac{V_w}{\rho} \cos(\eta - \chi + \psi). \tag{19}$$

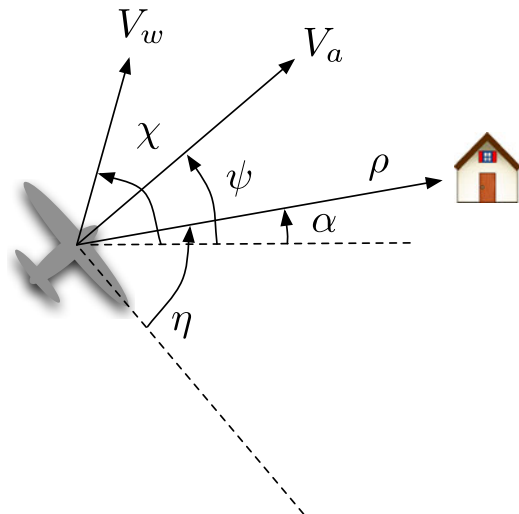
Note that throughout this example we use the mathematical convention of positive angles being measured anti-clockwise, whereas the navigation convention of measuring them clockwise is used in [21]. This accounts for differences in signs when comparing the texts. As noted above, we shall also make the standing assumption that $V_a > V_w$.

From the expressions 18 and 19, it can be shown [21] (see also Appendix) that as long as the wind speed V_w is not zero, pointing the camera right at a target will result in an elliptical orbit, while flying circular orbits will result in the target angle η oscillating, as illustrated in Fig. 5.

To balance this tradeoff we state the following control objectives, that were also used in [21], with $\rho_0 = 0$.

1. If $|\eta(0)| \leq \bar{\eta}$, ensure that $|\eta(t)| \leq \bar{\eta}$, for all $t \geq 0$, where $\bar{\eta}$ is the horizontal field of view of the camera.
2. Ensure that $|\phi(t)| \leq \bar{\phi}$, for all $t \geq 0$, where $\bar{\phi}$ is the maximum allowable roll angle.
3. Keep the stand-off distance to the target $\rho(t)$ close to some desired ρ_0 that represents a tradeoff between resolution and roll angle constraints.

Fig. 4 Polar coordinates for expressing the target position in the UAV frame. V_a, ψ are airspeed and heading, V_w, χ are wind magnitude and direction, α is absolute target direction, ρ is target distance and η is target direction relative to the wing mounted camera



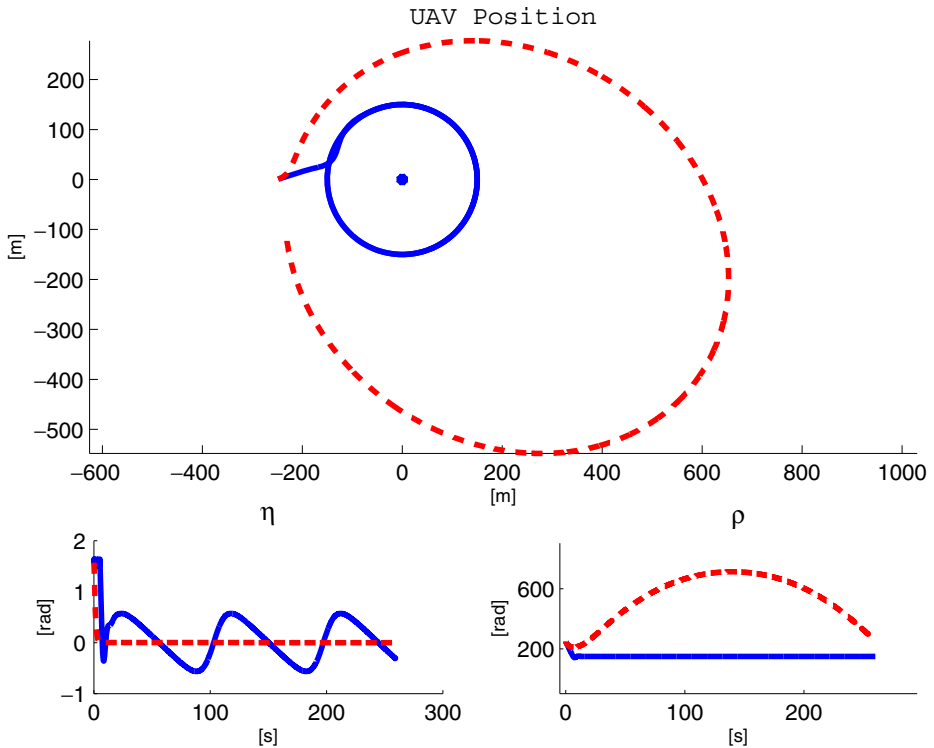


Fig. 5 Illustration of the two different control objectives. The *dashed curve* is given by the controller $u = u_1$, driving the target view angle down to $\eta = 0$. The *solid curve* is given by the controller $u = u_2$, driving the target distance to $\rho = \rho_0 = 150$ m

Note that the objectives above correspond to the last two objectives in Fig. 1. The first four were addressed a ground robot setting in [12].

In [21], an elegant solution to this problem was proposed in the form

$$u_S = -\frac{V_a}{\rho} \cos(\eta) - \frac{V_w}{\rho} \cos(\eta - \chi + \psi) + k_S \eta - v, \tag{20}$$

$$v = \operatorname{argmin}_{|v| \leq \min\{M(t), k_S \theta \bar{\eta}\}} \dot{W}_2, \tag{21}$$

where u_S is the control proposed by Saunders and Beard and a number of parameters are used, $k_S > 0$, $\theta \in (0, 1)$, $\lambda > 0$, $k_1 > 0$, together with the functions

$$W_2 = \frac{1}{2}(\rho - \rho_0 + \lambda \dot{\rho})^2, \\ M(t) = \frac{g}{V_a} \tan \bar{\phi} - k_1 |\eta(t)| - \frac{V_a + V_w}{\rho}.$$

As will be seen in Figs. 8–11, the controller in [21] achieves all objectives above. However, it is somewhat conservative, which has a negative effect on the performance relative to the ρ -objective. This might not be a problem when $\rho_0 = 0$,

as in [21], but when $\rho_0 \neq 0$, slightly better performance can be achieved even with a modular controller, as will be shown below.

To apply the scheme outlined in Section 4, we first need to formulate a Multi Objective Control Problem, as stated in Problem 1.

Problem 2 UAV Multi Objective Control Problem Let the state vector x be given by $x = (\rho, q, \eta, \psi) \in \mathcal{M}$ where

$$\mathcal{M} = (0, \infty) \times [-V_a - V_w, V_a + V_w] \times S^1 \times S^1, \tag{22}$$

and S^1 is the unit circle in the plane. Let the state q be defined by

$$q = \dot{\rho} \tag{23}$$

which makes

$$\dot{q} = uV_a \cos \eta + \frac{1}{\rho} (V_a \cos \eta + V_w \cos(\eta - \chi + \psi))^2. \tag{24}$$

The dynamical system $\dot{x} = f(x) + g(x)u$ is thus given by the Expressions 17–19 and Eq. 24. Furthermore, let

$$V_1 = \frac{1}{2}\eta^2, \tag{25}$$

$$V_2 = \frac{1}{2}k_{21}(\rho - \rho_0)^2 + \frac{1}{2}q^2, \tag{26}$$

$$U_{\text{adm}} = \left\{ u : |u| \leq \frac{g}{V_a} \tan \bar{\phi} \right\}, \tag{27}$$

with $k_{21} > 0$, and let the bounds b_{ij} be given by Table 2.

As can be seen, the first control objective is addressed by V_1 , keeping $|\eta| \leq \bar{\eta}$ corresponds to $V_1 \leq \frac{\bar{\eta}^2}{2}$. The second control objective is captured in U_{adm} . The third control objective is addressed by V_2 , since having $\rho = \rho_0$ corresponds to $V_2 = 0$. We have also added an objective of keeping within visibility range ρ_{vis} of the target. The bound $\rho < \rho_{\text{vis}}$ now holds if $V_2 \leq k_{21} \frac{(\rho_{\text{vis}} - \rho_0)^2}{2}$. Letting the visibility be the first priority, the camera view angle constraint be the second, and the ideal viewing distance be the third, we get the entries listed in Table 2.

Having defined the problem, we must now design the individual controllers u_i , needed in Definition 1. Note that they are completely modular, and only geared towards their corresponding objective function V_i , without regard to either the bounds b_{ij} or the other objectives. The design choices are presented in the two Lemmas below.

Table 2 The priority table containing the b_{ij} values of the simulation examples in Section 5

b_{ij}	0	1	2	3	4
$V_1 \leq$	∞	∞	$\frac{\bar{\eta}^2}{2}$	$\frac{\bar{\eta}^2}{2}$	$-\infty$
$V_2 \leq$	∞	$k_{21} \frac{(\rho_{\text{vis}} - \rho_0)^2}{2}$	$k_{21} \frac{(\rho_{\text{vis}} - \rho_0)^2}{2}$	0	$-\infty$

Lemma 3 *The function V_1 is a CLF for the restricted control problem given by Eq. 19 and the control*

$$u_1 = -\frac{V_a}{\rho} \cos(\eta) - \frac{V_w}{\rho} \cos(\eta - \chi + \psi) + k_1\eta \tag{28}$$

with $k_1 > 0$ makes the range ρ stay bounded and bounded away from zero, and the target angle η converge to 0 as $t \rightarrow \infty$, provided $\rho(0)$ is sufficiently large and $|\eta(0)|$ is sufficiently small.

Proof Precise conditions and a proof are given in the [Appendix](#). The proof builds on the observation that differentiating V_1 with respect to time yields

$$\dot{V}_1 = \eta \left(-u - \frac{V_a}{\rho} \cos(\eta) - \frac{V_w}{\rho} \cos(\eta - \chi + \psi) \right) \tag{29}$$

so that substituting the control u_1 in Eq. 28 we get

$$\dot{V}_1 = -k_1\eta^2 \leq 0, \tag{30}$$

i.e. V_1 has the stated CLF property. □

Lemma 4 *The function V_2 is a CLF for the restricted control problem given by Eq. 18 and the control*

$$u_2 = -\frac{k_{21}(\rho - \rho_0) + k_{22}q}{V_a \cos \eta} - \frac{1}{\rho} \frac{(V_a \cos \eta + V_w \cos(\eta - \chi + \psi))^2}{V_a \cos \eta} \tag{31}$$

with $k_{21}, k_{22} > 0$ makes the range ρ converge to ρ_0 and the target angle η stay confined to $[-\bar{\eta}, \bar{\eta}]$ as $t \rightarrow \infty$, provided $\rho(0)$ is sufficiently large and $k_{22}, |\rho(0) - \rho_0|$ are sufficiently small.

Proof Also here, precise conditions and a proof are given in the [Appendix](#). The proof builds on the observation that differentiating with respect to time we get

$$\dot{V}_2 = k_{21}(\rho - \rho_0)q + q\dot{q} \tag{32}$$

so that

$$\begin{aligned} \dot{V}_2 &= q(k_{21}(\rho - \rho_0) + uV_a \cos \eta) \\ &\quad + \frac{q}{\rho} (V_a \cos \eta + V_w \cos(\eta - \chi + \psi))^2, \end{aligned} \tag{33}$$

and substituting the control u_2 in Eq. 31 we get

$$\dot{V}_2 = -k_{22}q^2 \leq 0,$$

i.e. V_2 has the stated CLF property. □

In all figures, unless explicitly stated otherwise, we have used the following problem parameters: $g = 9.82$, $\chi = 45^\circ$, $V_a = 13\text{m/s}$, $V_w = 7\text{m/s}$, $\rho_0 = 100\text{m}$, $\rho_{\text{vis}} = 400\text{m}$, $\bar{\phi} = 45^\circ$, $\bar{\eta} = 35^\circ$. The design parameters for the proposed controller

were: $k = 2$, $k_1 = 1$, $k_{21} = 1$, $k_{22} = 1$ and for the controller in [21] we used $\theta = 0.99$, $k_S = 0.1$, $\lambda = 5$ as we found no suggested values in the paper. Finally, the parameters of the arbitration module are given in Table 2.

Remark 7 Note that in this example the control u is one dimensional and $U^*(x)$ is an interval. Thus the optimal control u^* of Eq. 3 is either equal to u_1 or u_2 , or an endpoint of the interval $U^*(x)$.

Running the controllers, we get the results in Figs. 6 to 11. In Fig. 6, we illustrate how the proposed controller works its way up through the columns of Table 2. Three trajectories, corresponding to three different values for ρ_{vis} are shown. All three trajectories start out at $\bar{j}(x) = 0$, aiming towards the target. Then, when $V_2 \leq b_{21}$, implying $\rho < \rho_{vis}$, we get $\bar{j}(x) = 1$ and the controller aims at bringing the target into view. After a sharp left turn we have $V_1 \leq b_{12}$, implying $|\eta| < \bar{\eta} = 35^\circ$, and $\bar{j}(x) = 2$. The next objective is $V_2 \leq b_{23} = 0$, implying $\rho = \rho_0$ which is asymptotically achieved when the trajectories approach the circular pattern. Note that throughout the execution, the previously achieved objectives where given higher priority than the ones not yet achieved, for example, the $|\eta| < \bar{\eta}$ bound constrained the progress towards $\rho = \rho_0$ when $\bar{j}(x) = 2$. Note that to avoid chattering in u_2 , when $\rho \gg \rho_0$, $\eta \approx 90^\circ$, a standard sliding mode solution had to be added [6].

To illustrate the effect of different bounds $\bar{\eta}$, we start the system in a favorable position, with $\bar{j}(x) = 2$, and get the limit cycles of Fig. 7. The Figure shows four different executions corresponding to the bound $\bar{\eta} = 20^\circ, 25^\circ, 30^\circ, 35^\circ$. Note how the different bounds are kept in a predictable way, leading to different tradeoffs between target angle η and target distance ρ .

In order to compare the proposed controller Eq. 3, hereafter denoted the 2010 controller, with the previous one Eq. 20, hereafter denoted the 2008 controller, we run simulations with the bounds $\bar{\eta} = 10, 20, 30, 40$ degrees. Figure 8 shows that $|\eta| \leq 40$ degrees is enough for the 2010 controller to fly a nearly perfect circle in the

Fig. 6 The proposed controller for the parameter $\rho_{vis} = 600$ m (solid), 400 m (dashed) and 200 m (dot-dashed). Note how $\bar{j}(x)$ increases from 0 at the start, to 1 at the end of the first segments, to 2 at the end of the sharp left turn, and asymptotically approaches the circle where $\bar{j}(x) = 3$

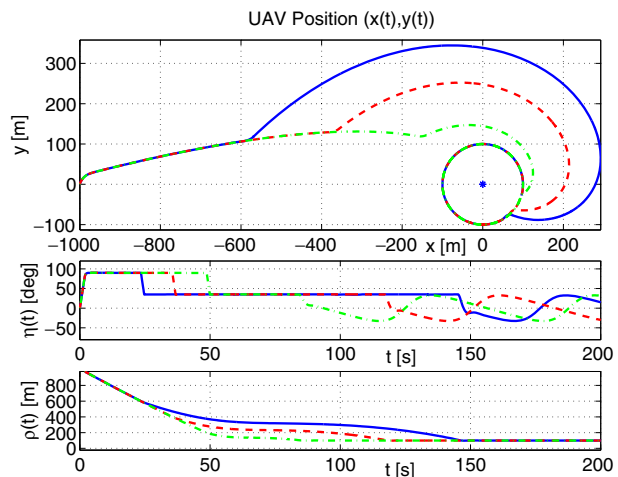
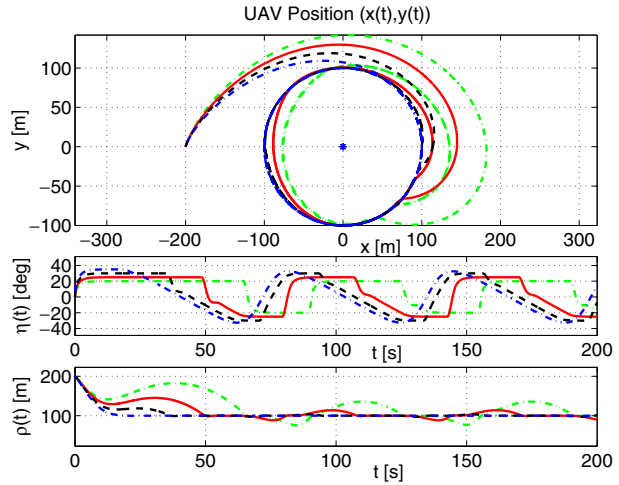


Fig. 7 The proposed controller for the bounds $\bar{\eta} = 20^\circ$ (dash-dotted green), 25° (solid), 30° (dashed), 35° (dash-dotted blue). The first and the last can be separated by context, or by comparison with Fig. 6



given wind while keeping the target in view, whereas the 2008 controller shows some difficulties in the downwind part of the loop.

In Fig. 9 we see that the $|\eta| \leq 30^\circ$ constraint does not permit a perfect circle for any of the controllers, but the 2010 controller manages to keep the deviations from ρ_0 small.

The trajectories in Fig. 10, with the bound $|\eta| \leq 20^\circ$ start to show resemblance to the $\eta = 0$ ellipse in Fig. 5, but the 2010 controller still manages to keep $\rho = \rho_0 = 100$ m on a short circular arc when passing north and south of the target. The 2008 controller does a good job, but produces slightly larger deviations from ρ_0 , as can be seen from the bottom plot.

Finally, setting the bound to $|\eta| \leq 10^\circ$ we get the plots of Fig. 11. Both controller produce similar ellipse-like trajectories with target distances oscillating around $\rho = \rho_0$.

Fig. 8 Comparison between the 2010 controller (solid) and the 2008 controller (dashed) [21], for $\bar{\eta} = 40^\circ$

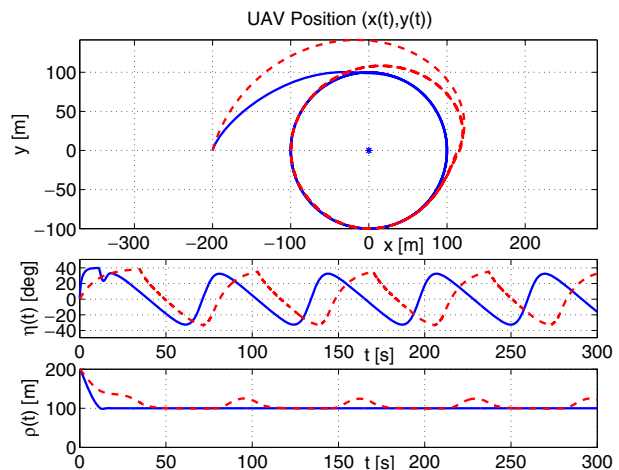


Fig. 9 Comparison between the 2010 controller (*solid*) and the 2008 controller (*dashed*) [21], for $\bar{\eta} = 30^\circ$

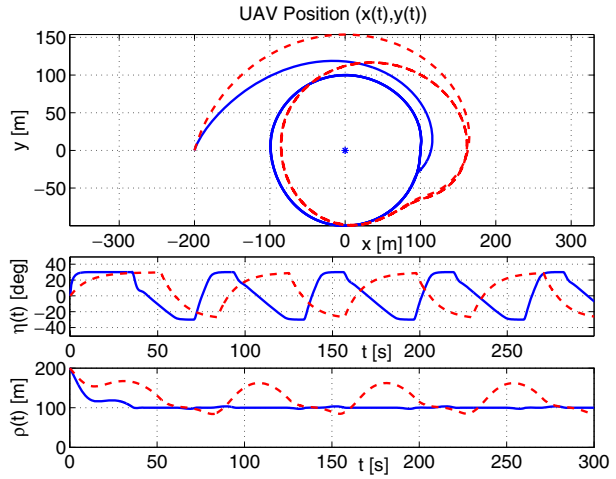


Fig. 10 Comparison between the 2010 controller (*solid*) and the 2008 controller (*dashed*) [21], for $\bar{\eta} = 20^\circ$

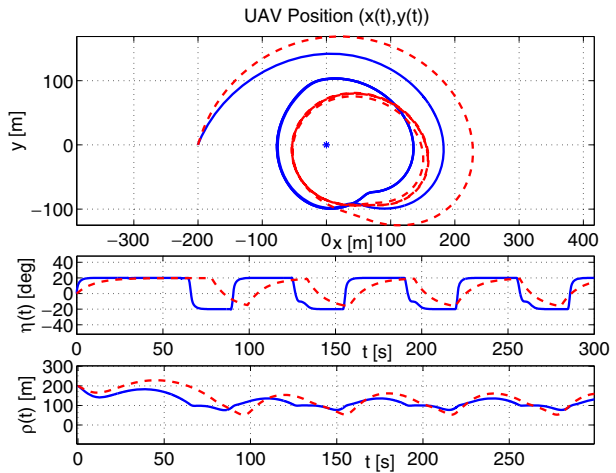
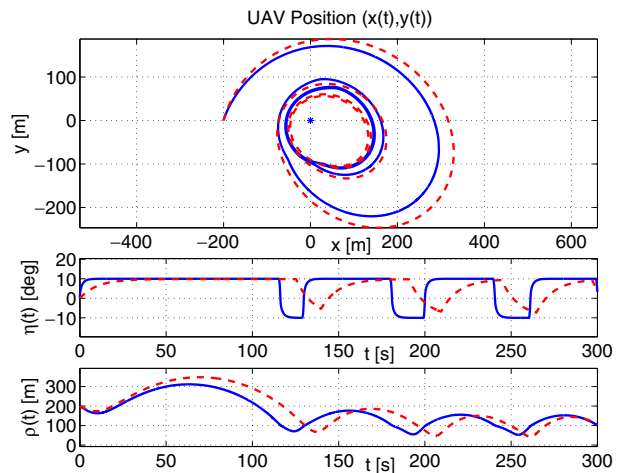


Fig. 11 Comparison between the 2010 controller (*solid*) and the 2008 controller (*dashed*) [21], for $\bar{\eta} = 10^\circ$



Remark 8 In the η -part of Fig. 11, we can see how the 2010 controller produces a bang-bang like behavior relative to the tight output constraints on η . This is probably performance effective, as it resembles the well known bang-bang phenomenon in input bounded optimal control [16]. Looking closer at the η -plot we see that the output is only approximately bang-bang, in fact, the bounds are approached exponentially. The parameter k , of Eq. 4 governs the speed of this exponential convergence, as discussed in Remark 6 above.

We conclude the simulation section with noting that in all the Figs. 8–11 the 2010 controller is very non-conservative. For example, in all simulations it immediately turns towards the target as tightly as the constraints allow, and thus gets a head start compared to the 2008 controller. We believe that this feature is advantageous in many Multi Objective Control Problems, not only the UAV example presented here.

6 Conclusions

Architectures affording modular design are attractive due to their flexibility and reusability, but can sometimes result in somewhat unpredictable system behavior. Model based designs on the other hand are less flexible, but might produce more predictable performance.

In this paper we have proposed an arbitration scheme that brings some of the predictability from the model based approaches into a modular architecture. This is done by keeping track of the time derivatives of a set of scalar functions representing the control objectives, in a way that is inspired by Control Lyapunov Functions (CLFs).

We were able to formally verify two properties of the proposed arbitration scheme. First, that objective violations only occur when they are necessary due to contradicting objectives. Second, that the system does indeed satisfy all objectives when no objective tradeoffs have to be made. These results are quite strong, considering the fact that the modular structure gives us no influence over what objectives and corresponding control modules are included in the framework, and insolvable problems are straightforward to design.

Finally, the approach was illustrated with a UAV surveillance example taken from the literature. Applying the framework we got a controller that performed equally, or better, than a model based approach tailored to the problem.

Appendix

Here we compare the proposed approach to the Dynamic Systems approach, as well as give proofs for Lemmas 3 and 4.

Relations to the Dynamic Systems Approach

The approach described in this paper is model based, and thus somewhat mathematical. A comparison to the most mathematical of the existing modular approaches, the

Dynamical Systems Approach [8] might therefore be appropriate, even though the latter is not, as we will see, model based.

In [8], a setting very similar to the one treated in [12] is investigated in terms of a navigation problem where a group of robots is to reach a goal location, avoid obstacles and stay within a prescribed distance of each other. In both papers, the outputs of the controller are desired heading and velocity. However, Large et al. focus on heading alone and let velocity be given by a function of the distance to nearest obstacle.

In [8], the controller producing the desired heading ϕ is defined by a dynamic system

$$(\dot{\phi}, \dot{w}_{\text{tar}}, \dot{w}_{\text{obs}}, \dot{w}_{\text{oth}}) = f(\phi, w_{\text{tar}}, w_{\text{obs}}, w_{\text{oth}}, u), \quad (34)$$

where $w_{\text{tar}}, w_{\text{obs}}, w_{\text{oth}}$ are weights and the ‘system input’ u is given by sensor data

$$u = (\psi_i, \Delta\psi_i, \psi_{\text{oth}}, r_{\text{oth}}, \text{noise}),$$

where ψ_i is direction to obstacles, $\Delta\psi_i$ is the angle occupied by the obstacle (an implicit distance measure), ψ_{oth} is the direction to the other robot, r_{oth} is distance to the other robot and *noise* is a noise term. The function f also includes a set of tuning parameters: $(R_i, R_{\text{agent}}, \sigma, d_0, \alpha_i, \gamma_{j,i}, b_1, b_2, b_3, d_1)$, where R_i is obstacle radius, R_{agent} is agent radius, σ is a safety distance, d_0 is a ‘radius where the agent begins to take obstacles into account’, α_i describes ‘the degree to which constraint i is appropriate to the agents’s current situation’, $\gamma_{j,i}$ describes ‘the extent to which constraint j is consistent with constraint i in the given current situation’, d_1 is the desired inter robot distance and b_1, b_2, b_3 are related to keeping this distance.

After giving a brief overview of the approach in [8], we now compare it to the approach proposed here. As can be seen, in [8] the user can specify safety distances σ and desired inter robot spacing d_1 but there is no way of analyzing how well these objectives are being met. In fact, the system Eq. 34 is evolving on a four-dimensional manifold (desired heading and 3 weights) that does not contain enough information to decide if the objectives (σ, d_1) are met or not. Furthermore, the approach is not model based, there is no model of what actually happens when the desired heading ϕ and velocity are set.

In comparison, the V_i of Problem 1 in the proposed approach are each an exact representation of the corresponding objective and give an instant measure of how well that objective is met. They furthermore give a notion of how those objectives are going to be affected by a given control signal, and the choice of control is based on this very notion. We refer to [12] for a detailed example of how a robot navigation problem would be addressed using the proposed approach.

To conclude, the similarities between [8] and the proposed approach are merely typographical, in that differential equations are used in both approaches. The underlying concepts are completely different.

Proof of Lemma 3

To prove the asymptotic stability of $\eta = 0$ we will use a standard result [20, Prop. 5.20], stating that a bounded trajectory has a compact ω -limit set and approaches that set as $t \rightarrow \infty$. Thus, if we show that the solution to the system is well defined,

bounded and that the controller never saturates, then we can use Eq. 30 to conclude that $\eta = 0$ holds for the ω -limit set that the trajectory converges to.

First, we will show that the trajectory is bounded and well defined. Note that by the definition of state manifold \mathcal{M} in Eq. 22, all components except ρ are bounded so we only need to show that the ρ -part is bounded. To that end, consider the transformed range $r \geq 0$ as

$$r = \rho \left(\cos \eta + \frac{V_w}{V_a} \cos(\eta - \chi + \psi) \right). \tag{35}$$

If

$$\bar{\eta} < \arccos(V_w/V_a) \tag{36}$$

then

$$\begin{aligned} \cos \eta + \frac{V_w}{V_a} \cos(\eta - \chi + \psi) &\geq \\ \cos \eta - \frac{V_w}{V_a} &\geq \cos \bar{\eta} - \frac{V_w}{V_a} > 0, \quad |\eta| \leq \bar{\eta}, \end{aligned} \tag{37}$$

and it is straightforward to see that

$$0 < \left(\cos \bar{\eta} - \frac{V_w}{V_a} \right) \rho \leq r \leq \left(1 + \frac{V_w}{V_a} \right) \rho, \quad |\eta| \leq \bar{\eta}, \tag{38}$$

and

$$\dot{r} = \frac{r \sin(\eta) u}{\cos \eta + \frac{V_w}{V_a} \cos(\eta - \chi + \psi)}, \quad |\eta| \leq \bar{\eta}. \tag{39}$$

Thus, when Eq. 36 holds, then r is bounded and bounded away from zero if and only if ρ is. In terms of r , the control u_1 in Eq. 28 reads

$$u_1 = -\frac{V_a}{r} \left(\cos \eta + \frac{V_w}{V_a} \cos(\eta - \chi + \psi) \right)^2 + k_1 \eta \tag{40}$$

and the (r, η, ψ) -part of the closed loop system becomes

$$\begin{aligned} \dot{r} &= -V_a \sin(\eta) \left(\cos \eta + \frac{V_w}{V_a} \cos(\eta - \chi + \psi) \right) \\ &\quad + \frac{rk_1 \sin(\eta) \eta}{\cos \eta + \frac{V_w}{V_a} \cos(\eta - \chi + \psi)} \end{aligned} \tag{41}$$

$$\dot{\eta} = -k_1 \eta. \tag{42}$$

$$\begin{aligned} \dot{\psi} &= -\frac{V_a}{r} \left(\cos \eta + \frac{V_w}{V_a} \cos(\eta - \chi + \psi) \right)^2 \\ &\quad + k_1 \eta. \end{aligned} \tag{43}$$

To show that u_1 and the solution to Eqs. 41–43 are well defined at all times we need to show that r stays away from zero. Let $t_1 > 0$ be a time such that the solution to Eqs. 41–43 is well defined over $[0, t_1]$ (such an interval exists by the continuity of the solution). During this interval, Eq. 42 has the solution $\eta(t) = \eta(0)e^{-k_1 t}$. When Eq. 36 holds then Eq. 37 holds and the last term on the right in Eq. 41 is positive so that if

$$|\eta(0)| \leq \bar{\eta} \tag{44}$$

we obtain the following estimate

$$\begin{aligned} r(t) - r(0) &= \int_0^t \dot{r}(s) ds \\ &\geq - \int_0^t V_a \sin(\eta(s)) \left(\cos(\eta(s)) + \frac{V_w}{V_a} \cos(\eta(s) - \chi + \psi(s)) \right) ds \\ &\geq - \int_0^t \left| V_a \sin(\eta(s)) \left(\cos(\eta(s)) + \frac{V_w}{V_a} \cos(\eta(s) - \chi + \psi(s)) \right) \right| ds \\ &\geq -(V_a + V_w) \int_0^t |\eta(s)| ds \\ &\geq -(V_a + V_w) \frac{|\eta(0)|}{k_1}, \quad t \in [0, t_1], \end{aligned}$$

using $\eta(t) = \eta(0)e^{-k_1 t}$ from above. Thus, if we define

$$r_{\min} = r(0) - (V_a + V_w) \frac{|\eta(0)|}{k_1}$$

then when Eq. 44 holds and

$$r_{\min} > 0 \tag{45}$$

we have $r(t) \geq r_{\min} > 0$ for all $t \in [0, t_1]$. However, r_{\min} is independent of t_1 and thus when Eqs. 36, 44 and 45 hold r cannot come arbitrarily close to zero. It follows that in this case r is bounded away from zero and the solution to Eqs. 41–43 is well defined at all times. The condition 45 will hold if $\rho(0)$ is sufficiently large and $\eta(0)$ is sufficiently small, respectively.

To establish an upper bound for r , we first note that we have $\dot{\rho} \leq V_a + V_w$ implying that $\rho(t) \leq \rho(0) + (V_a + V_w)t$, which with Eq. 38 gives $r(t) \leq 2(\rho(0) +$

$(V_a + V_w)t$). Then we assume that Eqs. 36, 44 and 45 hold, and since $\eta(t) = \eta(0)e^{-k_1 t}$ for all t the inequalities in Eq. 37 hold for all t and Eqs. 38 and 41 give the estimate

$$\begin{aligned} r(t) - r(0) &\leq \int_0^t |\eta(s)|(V_a + V_w) ds + \int_0^t \frac{|\eta(s)|r(s)k_1}{\cos \eta + \frac{V_w}{V_a} \cos(\eta - \chi + \psi)} ds \\ &\leq \int_0^t |\eta(0)e^{-k_1 s}|(V_a + V_w) ds \\ &\quad + \int_0^t \frac{|\eta(0)e^{-k_1 s}|2k_1(\rho(0) + (V_a + V_w)s)}{\cos \bar{\eta} - \frac{V_w}{V_a}} ds \leq K < \infty, \end{aligned}$$

for some constant K . Hence, $r(t)$ is also bounded for $t \geq 0$ in this case. It follows that when the control u_1 in Eq. 28 is applied and Eqs. 36, 44 and 45 hold the solution to the system (Eqs. 41–43) is well defined at all times and confined to a closed and bounded subset of the state manifold \mathcal{M} . Moreover, the ω -limit set of the system is then contained in the set

$$\{(\rho, q, \eta, \psi) \in \mathcal{M} : \eta = 0\}$$

and any trajectory approaches this set as $t \rightarrow \infty$. The only remaining issue is to ascertain that the control u_1 never saturates, i.e. that the conditions in Eq. 27 of the restricted control are satisfied at all times. This is easy, however, for Eq. 40 shows that when Eqs. 36, 44 and 45 hold then the maximal control requirements are bounded by

$$|u_1| \leq \frac{V_a}{r_{\min}} \left(1 + \frac{V_w}{V_a}\right)^2 + k_1 |\eta(0)|,$$

so by making r_{\min} sufficiently large and $|\eta(0)|$ small the bounds in Eq. 27 can always be satisfied. \square

We note in passing that Eq. 39 shows that when $\eta \equiv 0$ then r is constant so from Eq. 35 we see that ρ describes an ellipse.

Proof of Lemma 4

The closed loop system resulting from the control u_2 in Eq. 31 is

$$\dot{\rho} = q \tag{46}$$

$$\dot{q} = -k_{21}(\rho - \rho_0) - k_{22}q \tag{47}$$

$$\begin{aligned} \dot{\eta} &= \frac{k_{21}(\rho - \rho_0) + k_{22}q}{V_a \cos(\eta)} \\ &\quad + \frac{V_w}{\rho} \cos(\eta - \chi + \psi) \left(1 + \frac{V_w \cos(\eta - \chi + \psi)}{V_a \cos \eta}\right) \end{aligned} \tag{48}$$

$$\begin{aligned} \dot{\psi} &= -\frac{k_{21}(\rho - \rho_0) + k_{22}q}{V_a \cos(\eta)} \\ &\quad - \frac{1}{\rho} \frac{(V_a \cos(\eta) + V_w \cos(\eta - \chi + \psi))^2}{V_a \cos(\eta)}. \end{aligned} \tag{49}$$

The (ρ, q) -subsystem in Eqs. 46 and 47 is a linear second order system which is stable if $k_{21}, k_{22} > 0$. For any solution to Eqs. 46 and 47 we define ρ_{\min} and d_{\max} by

$$\rho_{\min} = \min_{t \geq 0} \{\rho(t) : q(0) \in [-V_a - V_w, V_a + V_w]\},$$

$$d_{\max} = \max_{t \geq 0} \{|\rho(t) - \rho_0| : q(0) \in [-V_a - V_w, V_a + V_w]\},$$

and assume that

$$\rho_{\min} > 0. \tag{50}$$

(This way, ρ_{\min}, d_{\max} are functions of $\rho(0)$ and ρ_0 only, parameterized by k_{21}, k_{22} .) Next, using Eqs. 18, 23 and some trigonometric manipulations we can write the η -dynamics in Eq. 48 as

$$\dot{\eta} = A(\rho, \eta, \psi, \chi) \tan(\eta) + B(\rho, \eta, \psi, \chi), \tag{51}$$

where

$$A(\rho, \eta, \psi, \chi) = \frac{V_w^2}{V_a \rho} \sin(\psi - \chi)^2 \sin(\eta) - k_{22} \left(1 + \frac{V_w}{V_a} \cos(\psi - \chi) \right),$$

$$B(\rho, \eta, \psi, \chi) = \frac{k_{21}(\rho - \rho_0)}{V_a \cos(\eta)} - k_{22} \frac{V_w}{V_a} \sin(\psi - \chi)$$

$$- \frac{V_w}{\rho} \left(1 + 2 \frac{V_w}{V_a} \cos(\psi - \chi) \right) \sin(\psi - \chi) \sin(\eta)$$

$$+ \frac{V_w}{\rho} \left(1 + \frac{V_w}{V_a} \cos(\psi - \chi) \right) \cos(\psi - \chi) \cos(\eta).$$

Now, if $\bar{\eta} < \pi/2$ and we can show that $\dot{\eta} > 0$ at $\eta = -\bar{\eta}$ and $\dot{\eta} < 0$ at $\eta = \bar{\eta}$ for all $t \geq 0$ then η can never escape the interval $[-\bar{\eta}, \bar{\eta}]$ and the solution to Eqs. 46–49 will be well defined for all $t \geq 0$. To this end we note that if

$$k_{22} \left(1 - \frac{V_w}{V_a} \right) > \frac{V_w^2}{V_a \rho_{\min}} \tag{52}$$

then we always have

$$A(\rho, \eta, \psi, \chi) \leq \frac{V_w^2}{V_a \rho} - k_{22} \left(1 - \frac{V_w}{V_a} \right)$$

$$\leq \frac{V_w^2}{V_a \rho_{\min}} - k_{22} \left(1 - \frac{V_w}{V_a} \right) < 0.$$

Further, if Eq. 52 is satisfied and moreover

$$- \left(\frac{V_w^2}{V_a \rho_{\min}} - k_{22} \left(1 - \frac{V_w}{V_a} \right) \right) |\tan(\bar{\eta})|$$

$$> \frac{1}{V_a} \left(k_{21} \frac{d_{\max}}{\cos(\bar{\eta})} + k_{22} V_w \right) + 2 \frac{V_w}{\rho_{\min}} \left(1 + 2 \frac{V_w}{V_a} \right) \tag{53}$$

then for $|\eta| \leq \bar{\eta}$ we have

$$\begin{aligned} B(\rho, \eta, \psi, \chi) &\leq \frac{1}{V_a} \left(k_{21} \frac{|\rho - \rho_0|}{\cos(\bar{\eta})} + k_{22} V_w \right) + 2 \frac{V_w}{\rho} \left(1 + 2 \frac{V_w}{V_a} \right) \\ &\leq \frac{1}{V_a} \left(k_{21} \frac{d_{\max}}{\cos(\bar{\eta})} + k_{22} V_w \right) + 2 \frac{V_w}{\rho_{\min}} \left(1 + 2 \frac{V_w}{V_a} \right) \\ &< - \left(\frac{V_w^2}{V_a \rho_{\min}} - k_{22} \left(1 - \frac{V_w}{V_a} \right) \right) |\tan(\bar{\eta})| \leq |A(\rho, \eta, \psi, \chi)| |\tan(\bar{\eta})|, \end{aligned}$$

so the term involving $\tan(\eta)$ on the right in Eq. 51 (strictly) dominates the other term. Summing up, when $|\eta(0)| \leq \bar{\eta} < \pi/2$ and the conditions Eqs. 50, 52 and 53 are satisfied then the solution to Eqs. 46–49 is well defined at all times and confined to a closed and bounded subset of the state manifold \mathcal{M} . Moreover, the ω -limit set of the system is then contained in the set

$$\{(\rho, q, \eta, \psi) \in \mathcal{M} : \rho = \rho_0, \eta \in [-\bar{\eta}, \bar{\eta}]\}$$

and any trajectory approaches this set as $t \rightarrow \infty$. As for the additional conditions imposed by the bounds in Eq. 27 on the control u_2 in Eq. 31 we note that the control requirements can be bounded as

$$|u_2| \leq \frac{k_{21} d_{\max} + k_{22}(V_a + V_w) + \frac{1}{\rho_{\min}}(V_a + V_w)^2}{V_a \cos(\bar{\eta})}$$

so by making d_{\max}, k_{22} sufficiently small and ρ_{\min} sufficiently large the bounds in Eq. 27 can always be satisfied. □

Acknowledgement This work was supported by Swedish Defence Research Agency (FOI) project E20674.

Open Access This article is distributed under the terms of the Creative Commons Attribution Noncommercial License which permits any noncommercial use, distribution, and reproduction in any medium, provided the original author(s) and source are credited.

References

1. Arkin, R.C.: Behavior-Based Robotics. MIT Press, Cambridge, MA (1998)
2. Artstein, Z.: Stabilization with relaxed controls. *Nonlinear Anal.* **7**(11), 1163–1173 (1983)
3. Balch, T., Arkin, R.: Behavior-based formation control for multirobot teams. *IEEE Trans. Robot. Autom.* **14**(6), 926–939 (1998)
4. Brooks, R.: A robust layered control system for a mobile robot. *IEEE J. Robot. Autom.* **2**(1), 14–23 (1986)
5. Brooks, R.: Elephants don't play chess. *Robot. Auton. Syst.* **6**(1–2), 3–15 (1990)
6. Khalil, H.K., Grizzle, J.W.: *Nonlinear Systems*. Prentice hall Englewood Cliffs, NJ (1996)
7. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. *Int. J. Rob. Res.* **5**(1), 90 (1986)
8. Large, E.W., Christensen, H.I., Bajcsy, R.: Scaling the dynamic approach to path planning and control: Competition among behavioral constraints. *Int. J. Rob. Res.* **18**(1), 37–58 (1999)
9. Lausen, H., Nielsen, J., Nielsen, M., Lima, P.: Model and behavior-based robotic goalkeeper. *Robocup 2003: Robot Soccer World Cup VII* (2004)
10. Maes, P.: Situated agents can have goals. *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*, pp. 49–70 (1990)

11. Mataric, M.J.: Behaviour-based control: examples from navigation, learning, and group behaviour. *J. Exp. Theor. Artif. Intell.* **9**(2–3), 323–336 (1997)
12. Ögren, P.: Improved predictability of reactive robot control using Control Lyapunov Functions. *IEEE Conference on Intelligent Robotics and Systems, IROS '08*, pp. 1274–1279 (2008)
13. Ögren, P., Leonard, N.E.: Obstacle avoidance in formation. In: *Proceedings of the IEEE International Conference on Robotics and Automation, ICRA*, vol. 2 (2003)
14. Olfati-Saber, R.: Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Trans. Automat. Contr.* **51**(3), 401–420 (2006)
15. Payton, D.W., Keirse, D., Kimble, D.M., Krozel, J., Rosenblatt, J.K.: Do whatever works: a robust approach to fault-tolerant autonomous control. *Appl. Intell.* **2**(3), 225–250 (1992)
16. Pinch, E.R.: *Optimal Control and the Calculus of Variations*. Oxford University Press, USA (1995)
17. Pirjanian, P., Christensen, H.I., Fayman, J.A.: Application of voting to fusion of purposive modules: an experimental investigation. *Robot. Auton. Syst.* **23**(4), 253–266 (1998)
18. Rosenblatt, J.K., Payton, D.: A fine-grained alternative to the subsumption architecture for mobile robot control. In: *Proceedings of the IEEE/INNS International Joint Conference on Neural Networks*, vol. 2, pp. 317–324 (1989)
19. Saffiotti, A., Konolige, K., Ruspini, E.H.: A multivalued logic approach to integrating planning and control. *Artif. Intell.* **76**(1–2), 481–526 (1995)
20. Sastry, S.: *Nonlinear Systems: Analysis, Stability, and Control*. Springer-Verlag, New York (1999)
21. Saunders, J., Beard, R.: Tracking a target in wind using a micro air vehicle with a fixed angle camera. In: *American Control Conference*, pp. 3863–3868 (2008)
22. Schmidt, D., Luksch, T., Wettach, J., Berns, K.: Autonomous behavior-based exploration of office environments. *3rd International Conference on Informatics in Control, Automation and Robotics* (2006)
23. Simmons, R.G.: Structured control for autonomous robots. *IEEE Trans. Robot. Autom.* **10**(1), 34–43 (1994)
24. Tanner, H.G.: Flocking with obstacle avoidance in switching networks of interconnected vehicles. *IEEE International Conference on Robotics and Automation, ICRA*, vol. 3 (2004)
25. Werger, B.B.: Cooperation without deliberation: a minimal behavior-based approach to multi-robot teams. *Artif. Intell.* **110**(2), 293–320 (1999)