Supervisory Control for Collision Avoidance in Vehicular Networks with Imperfect Measurements

Eric Dallal, Alessandro Colombo, Domitilla Del Vecchio and Stéphane Lafortune

Abstract— **We consider the problem of collision avoidance at road intersections in vehicular networks in the presence of uncontrolled vehicles, a disturbance, and measurement uncertainty. Our goal is to construct a supervisor of the continuous time system that is safe (i.e., avoids collisions), non-blocking (i.e., all vehicles eventually cross the intersection), and maximally permissive with respect to the discretization, despite the presence of a disturbance and of measurement uncertainty. We proceed in four steps: defining a discrete event system (DES) abstraction of the continuous time system, using uncontrollable events to model the uncontrolled vehicles and the disturbance; translating safety and non-blocking requirements to the DES level; solving at the DES level; and translating the resulting supervisor back from the DES level to the continuous level. We give sufficient conditions for this procedure to maintain the safety, non-blocking and maximal permissive properties as the supervisor is translated back from the DES level to the continuous level. Prior work on this problem based on similar abstractions assumes perfect measurement of position. Our method for handling measurement uncertainty is to introduce measurement events into the DES abstraction and then to compute the observer of the DES abstraction and the supremal controllable solution of the DES supervisory control problem.**

I. INTRODUCTION

The widespread diffusion of sensors and embedded computational and communication resources in production vehicles provides an opportunity to reduce road accidents, through the design of driver assist systems that can intervene to avert collisions, by means of warning signals or by temporarily taking control of the vehicle. In a control theoretic perspective, collision avoidance algorithms must satisfy two main constraints: first, given that lives are at stake, the control laws must be provably correct systems (safety property); second, the control systems must act only when strictly necessary (maximal permissiveness property). The challenge is magnified by the complexity of the environment in which the controllers have to act, which typically comprises uncontrolled agents (such as bikers or legacy vehicles), complex road topologies (multi-lane intersections or multiple nearby intersections and mergings), measurement and actuators noise, and uncertainties in the model equations.

A variety of approaches for collision avoidance of vehicles in a complex environment have been proposed in the literature. Common approaches include the computation of a maximal controlled invariant set [8], [14], [3], in a framework akin to verification of hybrid systems [13]. Recent improvements in such approaches have provided efficient algorithms that easily handle hundreds of vehicles, but cannot yet deal with uncontrolled agents or complex, realistic road topologies [3]. More flexible results are provided in [9], [1], which propose an algorithmic approach to collision avoidance, essentially structured as a decision tree. These approaches are flexible, but due to their complexity their performance is hard to assess. A set of algorithms that lie in between these two extremes is obtained using discrete abstractions, that is, discrete and finite representations of the system's dynamics, as was done in [4], [5], [7], [6]. By lifting the control problem to the discrete event level, most of the complexity stemming from the composite control specifications can be tackled in a relatively simple way at the discrete level, while at the same time the restrictiveness of the resulting controller can be measured in terms of the size of the allowed execution space, compared with the set of all possible executions.

In this paper, we extend the abstraction techniques that were used in [7] to handle the presence of measurement uncertainties. Typically, state uncertainties are handled by means of a *state estimator*, that is, a map that associates to a set of present and past measurements a possible set of current state values. Here, we leverage the discrete abstraction approach by solving the estimation problem at the discrete level. This simplifies the design and implementation of the estimator, since its domain is the finite dimensional set of the discrete abstraction. Moreover, we prove that the obtained estimator is the abstraction of an optimal continuous estimator (i.e., one providing the smallest state estimate compatible with the available information). Thus, we prove that abstraction and estimation form a commuting diagram. The resulting algorithm can handle multiple vehicles, input and measurement uncertainties, and uncontrolled agents.

The main contributions of this paper are as follows. We present a new abstraction method for the continuous dynamics of the vehicles under measurement uncertainty that results in a discrete event automaton with three disjoint sets of events: (i) controllable and observable control actions; (ii) observable but uncontrollable measurements; and (iii) uncontrollable and unobservable events for the uncontrolled vehicles and the disturbance inputs. In order to establish that the observer (aka, state estimator) of this discrete automaton is the "right" abstraction of an optimal continuous estimator based on the continuous dynamics and the measurement equation, we introduce the notions of *state reduction* and

Research supported in part by NSF grant CNS-0930081.

E. Dallal and S. Lafortune are with the EECS Dept. at the University of Michigan, MI, USA [edallal,stephane]@umich.edu

A. Colombo is with the DEIB at Politecnico di Milano, Italy alessandro.colombo@polimi.it

D. Del Vecchio is with the ME Dept. at MIT, MA, USA ddv@mit.edu

exact state reduction, which are akin to the notions of simulation/alternating simulation and alternating bisimulation in the hybrid systems literature, but adapted to the specific context under consideration in this paper. These new notions are then leveraged for proving that the solution at the abstracted DES level, when implemented at the continuous level, does meet the three requirements of safety, non-blockingness, and maximal permissiveness. Under our set-up, the DES problem can be solved on the observer of the discrete automaton, by computing the supremal controllable sublanguage.

The remainder of this paper is organized as follows. In Section II we describe a model of the system we analyze (Section II-A) and summarize our solution method from [7] in the case of perfect measurement (Section II-B). In Section III we define the notion of a state reduction and prove that when one system is a state reduction of the other, we can solve for a maximally permissive, safe, and non-blocking supervisor for the reduced system and use it to obtain a supervisor for the other system that is also safe, non-blocking, and maximally permissive. In Section III-B, we show that the DES abstraction we define in Section II-B is a state reduction of the system described in Section II-A. We formally describe the problem we wish to solve in subsection IV-A, define a new abstraction in Section IV-B, and show that this abstraction is a state reduction of a continuous estimator system in Section IV-C, thus allowing us to apply the same solution method for the case of imperfect measurement, but using this new abstraction instead. Finally, we conclude in Section V. Due to space constraints, intermediate results and proofs have been omitted. They are available from the authors.

II. MODEL & SUMMARY OF PERFECT INFORMATION CASE

A. Model

Consider a set $\mathcal{N} = \{1, \dots, n\}$ of vehicles, where $n = |\mathcal{N}|$. The vehicles are modeled as kinematic entities (integrators) and their collective dynamics are described by:

$$\dot{x} = v + d \tag{1}$$

where $x \in X \subset \mathbb{R}^n$ is the state, $v \in V \subset \mathbb{R}^n$ is the control input, and $d \in D \subset \mathbb{R}^n$ is a disturbance input representing unmodeled dynamics (for instance, the dynamic response of the vehicle to the engine torque). Assume that X is compact. We discretize the set of allowed inputs, assuming that $v \in V$ is a vector with elements in the finite set $\{\mu a, \mu(a+1), \dots, \mu b\}$, with $a, b \in \mathbb{N}$ and $\mu \in \mathbb{R}_+$, and that $d \in D = [d_{min}, d_{max}]^n$, with the vector $[0, \dots, 0] \in [d_{min}, d_{max}]^n$. We refer to $a\mu$ and $b\mu$ as v_{min} and v_{max}, respectively. We allow the possibility that a subset of the vehicles cannot be controlled. To represent this, we partition the vector v into two subvectors, $v_c \in V_c$ and $v_{uc} \in V_{uc}$, where v_c represents the control inputs of the controlled vehicles, whereas v_{uc} represents the control inputs of the uncontrollable vehicles, such that $v = (v_c, v_{uc})$ and $V = V_c \times V_{uc}$. Assume also that $v_{min} + d_{min} \geq \mu$, so that μ constitutes a lower bound on the velocity of the vehicles. Finally, assume that the input v is kept constant over time intervals $[k\tau, (k+1)\tau]$, and discretize the above system in time, with step τ, obtaining

$$x_{k+1} = x_k + u_k + \delta_k \tag{2}$$

with $x_k = x(k\tau)$, $u_k = v(k\tau)\tau$, $\delta_k = \int_{k\tau}^{(k+1)\tau} d(t)dt$. Calling $U = V\tau$ and $\Delta = D\tau$, we have that $u \in U$ and $\delta \in \Delta$. As for the set U, we write $U = U_c \times U_{uc}$, where U_c is the set of available actions for the controllable vehicles and U_{uc} is the set of actions of the uncontrollable vehicles. We use the notation $u = (u_c, u_{uc})$ to denote the actions of the controllable and uncontrollable vehicles for any $u \in U$.

We assume that the state x_k is measured imperfectly. The measured state at time $k\tau$ is

$$\chi_k := x_k + e, \tag{3}$$

with $\|e\|_\infty \leq e_{max}$.

Define a set Π_k for each road, and say that vehicle $i \in \Pi_k$ if vehicle i drives along road k. Describe the length of the portion of each road that belongs to the intersection as an interval $[\alpha_k, \beta_k] \subset \mathbb{R}$, and define a safety distance $\gamma \in \mathbb{R}_+$, common to all vehicles. We say that two vehicles $i \in \Pi_k$, $j \in \Pi_l$ with $k \neq l$ undergo a collision whenever $x_i \in [\alpha_k, \beta_k]$ and $x_j \in [\alpha_l, \beta_l]$ simultaneously. Similarly, we say that two vehicles $i, j \in \Pi_k$ undergo a collision whenever $|x_i - x_j| < \gamma$, $x_i \leq \beta_k$ and $x_j \leq \beta_k$. The subset of X of all collision points is called the *bad set* B. A trajectory $x(t)$ of (1) is ϵ*-safe* provided

$$\inf_{t \geq 0, b \in B} \|x(t) - b\|_\infty \geq \epsilon.$$

Define a set of discrete states $\tilde{Q}$ and a mapping $\ell : X \to \tilde{Q}$ from continuous to discrete states as follows:

$$\ell_i(x_i) := \begin{cases} \begin{array}{c} c\tau\mu, \text{ for } c \in \mathbb{N} \text{ s.t.} \\ c\tau\mu - \tau\mu/2 < x_i \leq c\tau\mu + \tau\mu/2 \end{array} & x_i \leq \beta_k \\ q_{i,m} & x_i > \beta_k \end{cases} \tag{4}$$

where k is the index of vehicle i's road ($i \in \Pi_k$). Define $\ell(x)$ as the vector $(\ell_1(x_1), \dots, \ell(x_n))$ and let $\ell(I) = \cup_{x \in I} \ell(x)$, for any $I \subseteq X$. In words, the space X is covered by a regular lattice with spacing $\tau\mu$ and vehicles before the end of the intersection are mapped to a discrete state corresponding to such a lattice point. Vehicles after the end of the intersection are mapped to "special" states $q_{i,m}$. The single discrete state $q_m = (q_{1,m}, \dots, q_{n,m})$ corresponds to the (unique) discrete state where all vehicles have crossed the intersection. Assume that, for all $q \in \tilde{Q}$, there exists some $x \in X$ such that $\ell(x) = q$.

B. Summary of Perfect Information Case

Let X/ℓ denote the quotient set of X by the equivalence classes induced by ℓ. In the case of perfect measurement, our aim is to design a supervisor $\sigma : X/\ell \to 2^{V_c}$ for (1) that enforces 0-safety, where $V_c = U_c/\tau$. More precisely, the following problem needs to be solved.

Problem 2.1: Given X/ℓ, define a supervisor that associates to each $x(k\tau) \in X$ a set of inputs $v_c \in V_c$ allowed for the interval $[k\tau, (k+1)\tau]$ and constant over this time interval, with the following properties:

- if $v_c(t) \in \sigma(x(\lfloor t/\tau \rfloor \tau))$ for $t \in [k\tau, (k+1)\tau]$, then $x(t)$ is 0-safe in the same time interval (0-safety)
- if $\sigma(x(k\tau)) \neq \emptyset$, $v_c(t) \in \sigma(x(\lfloor t/\tau \rfloor \tau))$ for $t \in [k\tau, (k+1)\tau]$, and $\ell(x((k+1)\tau)) \neq q_m$, then $\sigma(x((k+1)\tau)) \neq \emptyset$ (non-blockingness)
- if $\tilde{\sigma} \neq \sigma$ and $\tilde{\sigma}$ satisfies the two properties above, then $\tilde{\sigma}(x(k\tau)) \subseteq \sigma(x(k\tau))$ for all $k \geq 0$ (maximal permissiveness).

In [7], we solve this problem in five steps: defining a suitable DES abstraction of the continuous time system; modeling the disturbance and the uncontrolled vehicles as uncontrollable events; translating the specifications of Problem 2.1 from the continuous to the DES level; solving for the maximally permissive, safe, and non-blocking supervisor at the DES level; and translating the solution back to the continuous domain. Following is a brief summary of the DES abstraction and of some of the significant results in [7]. This abstraction is repeated here as it will form the basis for the new results in this paper for the imperfect measurement case.

The DES abstraction G consists of an automaton defined over the state $\tilde{Q}$, with three categories of events: the set of controllable events U_c, corresponding to the actions of the controlled vehicles; the set of uncontrollable events U_{uc}, corresponding to the actions of the uncontrolled vehicles; and the set of uncontrollable events W, used to represent the "actions" of the disturbance. Physically, W represents a discretization of the set of disturbances Δ, defined by: $W = \{k\tau\mu : k \in \mathbb{Z} \wedge \lfloor \delta_{min}/(\tau\mu) \rfloor \leq k \leq \lceil \delta_{max}/(\tau\mu) \rceil\}^n$. Thus, for every $x \in X$, $u \in U$, $\delta \in \Delta$ and $q = \ell(x)$, there exists a $w \in W$ such that $q + u + w = \ell(x + u + \delta)$. Discrete event system G's transition function is defined simply as $\psi(q, u, w) = q + u + w$. With three different types of events, each of which must occur exactly once during each interval of time $[k\tau, (k+1)\tau]$, we define a three-layer transition structure, with events alternating between those of U_c, U_{uc}, and W. To define the discrete system state in between the occurrence of events in U_c and U_{uc} and in between the occurrence of events in U_{uc} and W (all of which occur simultaneously in the continuous-time system), we introduce two sets of "intermediate" states Q_{I1} and Q_{I2} (disjoint from each other and from $\tilde{Q}$ and with no physical meaning), and three intermediate transition functions: $\psi_1 : \tilde{Q} \times U_c \to Q_{I1}$, $\psi_2 : Q_{I1} \times U_{uc} \to Q_{I2}$, and $\psi_3 : Q_{I2} \times W \to \tilde{Q}$, defined only by $\psi(q, u_c, u_{uc}, w) = \psi_3(\psi_2(\psi_1(q, u_c), u_{uc}), w)$. We take the set of marked states to be the set $Q_m = \{q_m\}$. Finally, we define a set Q_0 of possible initial states, which we model by introducing a dummy initial state q^0 and having transitions from q^0 to each state $q \in Q_0$ with event label e_q. We denote this set of events by $E_Q := \{e_q : q \in Q_0\}$. The final DES is defined as:

$$G := (Q, U_c \times U_{uc} \times W \cup E_Q, \psi, q^0, Q_m) \qquad (5)$$

where $Q = \{q^0\} \cup \tilde{Q} \cup Q_{I1} \cup Q_{I2}$. It is shown in [7] that G simulates system (2) and that system (2) alternatingly simulates G. (See [12] for formal definitions of these notions.) We next proceed to translate the system requirements of Problem 2.1 to the DES domain. To translate the safety requirement, we define a transition from state q to $q' = \psi(q, u, w)$ as safe if and only if there do not exist $x(k\tau) \in X$ and $d(t)$ such that $\ell(x(k\tau)) = q$, $\ell(x(k\tau) + u + \int_{k\tau}^{(k+1)\tau} d(t)) = q'$ and $x(t)$ crosses the bad set at some time in the interval $[k\tau, (k+1)\tau]$. See [7] for equations dependent on q, u, w, μ, τ, and B for verifying when this condition is satisfied. The specification for G requires that all such transitions be avoided. To translate the non-blocking requirement, the specification for G requires that all executions must eventually reach a marked state, which means that all the vehicles must eventually cross the intersection, from the way that Q_m was defined.

The solution at the DES level requires the computation of the supremal controllable sublanguage of the specification with respect to $\mathcal{L}(G)$, a procedure which results in a supervisor $S : \tilde{Q} \to 2^{U_c}$ that is safe, non-blocking, and maximally permissive with respect to the specification, thus satisfying all the requirements. Translating back from the DES to the continuous level is achieved by taking the supervisor σ as:

$$\sigma(x(k\tau)) = \{u_c/\tau : u_c \in S(\ell(x(k\tau)))\}. \qquad (6)$$

In [7], we prove that the supervisor σ obtained by this procedure satisfies the requirements of Problem 2.1 and give algorithms for efficiently computing the DES supervisor S. In this work, we show how to modify the automaton G and the DES specification to solve the problem of collision avoidance in the presence of measurement uncertainty.

III. STATE REDUCTIONS AND EXACT STATE REDUCTIONS

In this paper, we do not use the simulation & alternating simulation relations of our previous work [7]. Instead, we define the notions of *state reduction* and *exact state reduction*. A state reduction can be thought of as similar to a simulation / alternating simulation relation and an exact state reduction can be thought of as similar to an alternating bisimulation relation. As we will see, the use of state reductions as opposed to simulation/alternating simulation relations allows us to obtain maximally permissive supervisors rather than merely safe and non-blocking solutions. We prove here that G is a state reduction of (2). This theorem is necessary as we will make use of it in proving the correctness of our approach in the case of imperfect measurement.

A. State Reductions and Supervisory Control

We begin by describing the state reduction relation below. In what follows, the notation $\mathbf{Post}_u(x)$ is the set of states reachable from state x given control decision u and is obtained by considering all possible uncontrollable events that may follow. For example, for DES G in equation (5), we would write $\mathbf{Post}_{u_c}(q) = \cup_{u_{uc} \in U_{uc}, w \in W} \psi(q, u_c, u_{uc}, w)$.

Definition 3.1 (State Reduction): Let a system S be defined as a tuple $S = (X, U, \to, Y, H)$, where X is the set of states, U is a set of control inputs, $\to \subseteq X \times U \times X$ is a transition relation, Y is an output set, and $H : X \to Y$ is the output function. Given two systems S_a and S_b with $Y_a = Y_b$, we say that S_a is a state reduction of S_b with state relation $R \subseteq X_a \times X_b$ and control relation $C \subseteq U_a(x_a) \times U_b(x_b)$ if:

1) R^{-1} is a function.
2) C is a bijection.
3) $H_a(x_a) = H_b(x_b)$ if and only if $(x_a, x_b) \in R$.
4) $\forall (x_a, u_a, x'_a) \in \rightarrow_a$, $\exists (x_b, u_b, x'_b) \in \rightarrow_b$ such that $(x_a, x_b) \in R$, $(u_a, u_b) \in C$, and $(x'_a, x'_b) \in R$.
5) $\forall (x_a, x_b) \in R$, $(u_a, u_b) \in C$ and $x'_b \in \mathbf{Post}_{u_b}(x_b)$, $\exists x'_a \in \mathbf{Post}_{u_a}(x_a)$ such that $(x'_a, x'_b) \in R$.

In words, condition 1) signifies that every $x_b \in X_b$ is in relation with exactly one $x_a \in X_a$, condition 5) signifies that, for every $(x_a, x_b) \in R$, $(u_a, u_b) \in C$ and $x'_b \in \mathbf{Post}_{u_b}(x_b)$, there exists $(x_a, u_a, x'_a) \in \rightarrow_a$ which models $(x_b, u_b, x'_b) \in \rightarrow_b$, and condition 4) signifies that every transition in $\rightarrow_a$ models *some* transition in $\rightarrow_b$. Significantly, conditions 4) and 5) can be achieved by construction for any system S_b, and relations R and C satisfying conditions 1), 2), and 3).

In the remainder of this paper we will often refer to the computation of a maximally permissive, safe, and non-blocking supervisor of a system at the DES level. Obtaining this supervisor consists of solving the basic supervisory control problem in the non-blocking case, or BSCP-NB, as described in [11], [2]. Specifically, problem BSCP-NB computes the supremal controllable sublanguage of a specification $\mathcal{L}_m(H)$ with respect to $\mathcal{L}(G)$, where G is a system automaton and H is a specification automaton. In general, the event set of G and H, denoted by E is is partitioned into controllable events E_c and uncontrollable events E_{uc}. The solution to problem BSCP-NB is the language $(\mathcal{L}_m(H))^{\uparrow C}$, where $\uparrow C$ denotes the supremal controllable sublanguage operation. The standard algorithm which solves this problem is given in [15] and constructs a supervisor S such that $\mathcal{L}_m(S/G) = (\mathcal{L}_m(H))^{\uparrow C}$ and $\mathcal{L}(S/G) = \overline{(\mathcal{L}_m(H))^{\uparrow C}}$, where S/G is the system G controlled by S and $\overline{L}$ denotes the *prefix closure* of language L, which is all the strings in L and all their prefixes.

In the theorem that follows (Thm. 3.1), we compute the maximally permissive, safe, and non-blocking supervisor of a system with transitions $\rightarrow_a \subseteq X_a \times U_a \times X_a$ with respect to a safety specification $Safe_a \subseteq \rightarrow_a$ and set of marked states $X_{m,a} \subseteq X_a$. Consistent with the above description of BSCP-NB, we need two automata, denoted by G_a and H_a, to capture the system behavior $\rightarrow_a$ and the legal behavior given by $Safe_a$ and $X_{m,a}$. Assume that we have an automaton G_a with states X_a and transition function ψ_{G_a} satisfying the following two conditions (where ! means is defined):

$$\psi_{G_a}(x_a, u_a)! \Leftrightarrow \exists x'_a \in X_a : (x_a, u_a, x'_a) \in \rightarrow_a \qquad (7)$$

$$\exists t \in E^*_{uc} : \psi_{G_a}(x_a, u_a t) = x'_a \Leftrightarrow (x_a, u_a, x'_a) \in \rightarrow_a \qquad (8)$$

In words, $(x_a, u_a, x'_a) \in \rightarrow_a$ means that controllable event u_a is defined from state x_a in G_a and there exists some uncontrollable sequence of events following u_a that takes G_a from $\psi_{G_a}(x_a, u_a)$ to x'_a. Given G_a, $Safe_a$ and $X_{m,a}$, we construct subautomaton $H_a \sqsubseteq G_a$ such that $X_{m,a} \subseteq X_a$ is the set of marked states and with transition function ψ_{H_a} satisfying conditions (7) and (8), but with $Safe_a$ instead of $\rightarrow_a$. We solve BSCP-NB for H_a and G_a. Because $H_a \sqsubseteq G_a$, S is of the form $S : X_a \rightarrow 2^{U_c}$.

The usefulness of Def. 3.1 is illustrated in the following theorem:

Theorem 3.1: Suppose that system S_a is a state reduction of S_b with state relation R and control relation C and that we are given a safety specification $Safe_b \subseteq \rightarrow_b$ and a set of marked states $X_{m,b} \subseteq X_b$ for system S_b. Suppose that the set of marked states satisfies the condition: $H_b(x_{b,1}) = H_b(x_{b,2}) \Rightarrow (x_{b,1} \in X_{m,b} \Leftrightarrow x_{b,2} \in X_{m,b})$. Define the safety specification $Safe_a \subseteq \rightarrow_a$ for system S_a by $(x_a, u_a, x'_a) \in Safe_a$ if and only if, for all $(x_b, u_b, x'_b) \in \rightarrow_b$ such that $(x_a, x_b) \in R$, $(u_a, u_b) \in C(x_a, x_b)$ and $(x'_a, x'_b) \in R$, we have that $(x_b, u_b, x'_b) \in Safe_b$. Define the set of marked states $X_{m,a} \subseteq X_a$ by $x_a \in X_{m,a}$ if and only if $\exists x_b \in X_{m,b}$ such that $(x_a, x_b) \in R$. Suppose that we have a maximally permissive, safe, and non-blocking supervisor $\sigma_a : Y \rightarrow 2^{U_a}$, where Y is the (common) output space. Define the supervisor $\sigma_b : Y \rightarrow 2^{U_b}$ by $u_b \in \sigma_b(y)$ iff $\exists u_a \in \sigma_a(y)$ such that $(u_a, u_b) \in C$. Then σ_b is safe, non-blocking, and maximally permissive among supervisors of the form $\sigma_b : Y \rightarrow 2^{U_b}$.

The above theorem shows that it is possible to compute a supervisor for a system with a large or infinite state space by abstracting that system to one with a finite state space, computing a supervisor for the reduced system, and translating back. Furthermore, this process conserves not only safety and non-blockingness in the translation, but also maximal permissiveness.

Next, we define an *exact* state reduction, which is akin to an alternating bisimilarity relation.

Definition 3.2 (Exact State Reduction): Given two systems S_a and S_b with $Y_a = Y_b$, we say that S_a is an exact state reduction of S_b with state relation $R \subseteq X_a \times X_b$ and control relation $C \subseteq U_a(x_a) \times U_b(x_b)$ if S_a is a state reduction of S_b and:

6) for every $(x_a, x_b) \in R$, for every $(u_a, u_b) \in C$ and for every $x'_a \in \mathbf{Post}_{u_a}(x_a)$, $\exists x'_b \in \mathbf{Post}_{u_b}(x_b)$ satisfying $(x'_a, x'_b) \in R$.

B. Properties of our Abstraction

In this paper, we have the same set of control decisions in both the continuous and discrete domains. Thus, the relation C will always be taken to be $C = \cup_{u_c \in U_c}(u_c, u_c)$. We prove that system G is a state reduction of (2) here and show that, additionally, it is an exact state reduction when δ_{min} and δ_{max} are multiples of μ.

Theorem 3.2: Define the observation maps $H_X(x) := \ell(x)$, $H_Q(q) := q$, and the relation $R := \{(x, q) \in X \times Q : \ell(x) = q\}$. Then system G is a state reduction of (2).

Theorem 3.3: Define H_X, H_Q, and R as in Thm. 3.2. If δ_{min} and δ_{max} are multiples of μ, then G is an exact state reduction of (2).

IV. CASE OF IMPERFECT MEASUREMENT

In this section, we show how to extend the results of [7] to obtain a maximally permissive, safe, and non-blocking supervisor in the presence of measurement error. We begin by describing a system defined over continuous state estimates that implements a prediction-correction scheme and proceed to formally describe the properties of the supervisor we wish to obtain, analogously to Prob. 2.1. We then describe how to modify the discrete abstraction G of equation (5) to deal with system measurements, which we do by first introducing a finite set of *observable* "measurement" events to our abstraction, and second by treating all the uncontrollable events as unobservable, resulting in a modified DES G'. Next, we construct $Obs(G')$, the observer of G' with respect to its unobservable events (see e.g., [2]), and show that it is a state reduction of the system defined over continuous state estimates. Finally, we define a safety and non-blocking specification for $Obs(G')$ and invoke Thm. 3.1 to show that we can obtain the desired supervisor in the case of imperfect measurement by computing the maximally permissive, safe, and non-blocking supervisor of $Obs(G')$ with respect the specification and translating this supervisor from the DES level to the continuous level.

A. Problem Description

Given a maximal error in measurement of e_{max}, define the function $L : X \to 2^X$ by $L(\chi) = [\chi - \mathbf{1}e_{max}, \chi + \mathbf{1}e_{max})$, where $\mathbf{1}$ denotes the vector $(1, \dots, 1)^T$ and, for any two vectors $a, b \in \mathbb{R}^n$, $[a, b)$ denotes the box $\{x \in \mathbb{R}^n : a_i \le x_i < b_i, i = 1, \dots, n\}$. We say that continuous state x is *consistent* with measurement χ if $x \in L(\chi)$. Now consider a set-membership estimation scheme for system (2) with observations (3). Let $I_l^p \subseteq X$ and $I_k^c \subseteq X$ be, respectively, the predicted state estimate, and the state estimate after correction. For a given controllable input $u_c \in U_c$, the evolution of the two sets is described by:

$$\begin{array}{rcl} I_0^p &=& \ell^{-1}(Q_0) = \{x \in X : \ell(x) \in Q_0\} \\ I_k^c &=& I^c(I_k^p, \chi_k) = I_k^p \cap L(\chi_k) \\ I_{k+1}^p &=& I^p(I_k^c, u_c) = \bigcup_{\delta \in \Delta, u_{uc} \in U_{uc}, x \in I_k^c} [x + u + \delta] \end{array} \tag{9}$$

Let $2^X/\ell$ denote the quotient set of 2^X by the equivalence classes induced by ℓ. In the case of imperfect measurement, our aim is now to design a supervisor $\sigma : 2^X/\ell \to 2^{V_c}$ for (9) that enforces 0-safety. More precisely, we aim to solve the following problem.

Problem 4.1: Given $2^X/\ell$, define a supervisor that associates to each $I_k^c \in 2^X$ a set of inputs $v_c \in V_c$ allowed for the interval $[k\tau, (k+1)\tau]$ and constant over this time interval, with the following properties:

- if $v_c(t) \in \sigma(I^c_{\lfloor t/\tau \rfloor \tau})$ for $t \in [k\tau, (k+1)\tau]$, then $x(t)$ is 0-safe in the same time interval (0-safety)
- if $\sigma(I_k^c) \neq \emptyset$, $v_c(t) \in \sigma(I^c_{\lfloor t/\tau \rfloor \tau})$ for $t \in [k\tau, (k+1)\tau]$, and $\ell(I_{k+1}^c) \neq \{q_m\}$, then $\sigma(I_{k+1}^c) \neq \emptyset$ (non-blockingness)
- if $\tilde{\sigma} \neq \sigma$ and $\tilde{\sigma}$ satisfies the two properties above, then $\tilde{\sigma}(I_k^c) \subseteq \sigma(I_k^c)$ for all $k \ge 0$ (maximal permissiveness).

B. The Observer

We assume that the continuous system begins operation with a single measurement χ of the current state. Furthermore, we take the events of U_{uc} (the actions of the uncontrolled vehicles) and W (the effect of the disturbance) to be unobservable. Thus, at any given time, we can define a notion of information state (see e.g., [10]), representing the set of states that the system could be in, given all past information (the sequence of chosen control decisions and position measurements). Now, we would like to introduce an observer of the state into our discrete event model, that uses past knowledge of the state and current measurement to estimate the current information state. We obtain the estimator by first defining a new event, denoted by λ, that represents the measurement. We begin by partitioning the set of possible measurements X into a set of equivalence classes Λ. Then, for any state $q \in \tilde{Q}$ and for any measurement $\chi \in X$, we define a transition $\psi^c(q, \lambda)$ if and only if some x such that $\ell(x) = q$ is consistent with the measurement χ, where $\lambda = [\chi]$, the equivalence class containing χ. We thus obtain a new transition structure, by composing $\psi(q, u, w)$ with the new transition $\psi^c(q, \lambda)$. We then take the observer of the DES with transitions $\psi(\cdot, u, w) \circ \psi^c(q, \lambda)$ and show that, with this four-layer transition structure, the observer correctly realizes a prediction-correction estimator.

Consider some information state $\iota \subseteq \tilde{Q}$. Based on the transition function ψ, the set $\psi(\iota, u, w) := \bigcup_{q \in \iota} \psi(q, u, w)$ is the set of all states that are reachable given initial state $q \in \iota$. In other words, this is the best *prediction* given our knowledge of the initial conditions and of the dynamics. We can correct the prediction once a measurement is taken, by defining a transition that sends $\psi(\iota, u, w)$ to a subset of itself, consisting of all the states that are consistent with the measurement. For any discrete state q, define $\ell^{-1}(q) := \{x \in X : \ell(x) = q\}$ and for any discrete information state $\iota \subseteq \tilde{Q}$, define $\ell^{-1}(\iota) := \cup_{q \in \iota} \ell^{-1}(q)$. We define $\psi^c(q, \chi)$ by:

$$\psi^c(q, \chi) = \begin{cases} q, & \ell^{-1}(q) \cap L(\chi) \neq \emptyset \\ \text{undefined}, & \text{else} \end{cases} \tag{10}$$

We extend this definition to any discrete information state $\iota \subseteq \tilde{Q}$ by defining $\psi^c(\iota, \chi) = \{q \in \iota : \ell^{-1}(q) \cap L(\chi) \neq \emptyset\}$. Because the set of possible measurements is X, which is infinite, we define a finite set of equivalence classes Λ. Specifically, we write $\chi_1 \equiv \chi_2$ if $\psi^c(\tilde{Q}, \chi_1) = \psi^c(\tilde{Q}, \chi_2)$. Denote by $[\chi]$ the equivalence class of χ and let Λ be the set of such equivalence classes. Then, for any $\lambda \in \Lambda$, define the *correction* $\psi^c(\iota, \lambda)$ as $\psi^c(\iota, \chi)$, for any χ such that $\lambda = [\chi]$. This is well defined since $\psi^c(\iota, \chi) = \psi^c(\tilde{Q}, \chi) \cap \iota$, and hence $\psi^c(\tilde{Q}, \chi_1) = \psi^c(\tilde{Q}, \chi_2) \Rightarrow \psi^c(\iota, \chi_1) = \psi^c(\iota, \chi_2)$.

We now write, with slight abuse of notation, $\psi(q, \lambda, u, w) := \psi(\cdot, u, w) \circ \psi^c(q, \lambda)$, which results in a four-layer transition structure. Note that, in order to enforce the property that the language of this new system should be a subset of $(\Lambda U_c U_{uc} W)^*$, we must introduce another layer of intermediate states, $\tilde{Q}'$, between $\tilde{Q}$ and Q_{I1}. This layer, which is reached upon the occurrence of a measurement λ, is a copy of $\tilde{Q}$. Thus, when we write $\psi^c(q, \lambda) = q$, we take the input state to be in $\tilde{Q}$ and the output state to be in $\tilde{Q}'$. Now, define the set of events Λ to be observable but uncontrollable, the set U_c to be observable and controllable and the two sets of events U_{uc} and W to be unobservable and uncontrollable. Let the resulting system be called G'. Finally, we compute the observer $Obs(G')$, which will have transition function $\bar{\psi} = \bar{\psi}^c \cup \bar{\psi}^p$, where $\bar{\psi}^c : 2^{\tilde{Q}} \times \Lambda \to 2^{\tilde{Q}'}$ and $\bar{\psi}^p : 2^{\tilde{Q}'} \times U_c \to 2^{\tilde{Q}}$. That is, the states of $Obs(G')$ are information states and its events are either control decisions or measurement events. We will define the marked states of $Obs(G')$ later on. By construction, all controllable events of G' are observable.

C. Solution Method

In this subsection, we demonstrate that, when G is an exact state reduction of system (2), $Obs(G')$ will be an exact state reduction of the continuous estimator system of (9). We proceed to give equations for the safety and non-blocking specifications of system (9). Finally, we invoke Thm. 3.1 to show that we can solve Prob. 4.1 by appropriately defining safety and non-blocking specifications on $Obs(G')$, computing the maximally permissive, safe, and non-blocking supervisor S of $Obs(G')$ with respect to these specifications, and translating this supervisor to the continuous domain to obtain the desired supervisor σ that solves Prob. 4.1.

Theorem 4.1: Define the observation maps $H_{2^X}(I) = \ell(I)$ and $H_{2^{\tilde{Q}}}(\iota) = \iota$ and the relation $R = \{(\iota, I) \in 2^{\tilde{Q}} \times 2^X : \iota = H_{2^X}(I)\}$. If G is an exact state reduction of (2), then $Obs(G')$ is also an exact state reduction of (9).

Remark 4.1: If G is an (inexact) state reduction of (2) then $Obs(G')$ will not in general be a state reduction of (9). The reason for this is that $(\iota, I) \in R$ does not imply $(\bar{\psi}^p(\iota, u_c), I^p(I, u_c)) \in R$ when G is not an exact state reduction of (2).

Next, we wish to apply Thms. 3.1 and 4.1 to solve Prob. 4.1. To do this, we must define the transition function $\to_{(9)}$, safety specification $Safe_{(9)}$, and set of marked states $I_{m,(9)}$ for system (9). Let $\to_{(2)}$ and $Safe_{(2)}$ denote the transition function and safety specification of system (2). Then:

$$\begin{gathered}(I, u_c, I') \in \to_{(9)} \subseteq 2^X \times U_c \times 2^X \text{ if} \\ \exists \chi \in X : I^c(I^p(I, u_c), \chi) = I'\end{gathered} \tag{11}$$

$$\begin{gathered}(I, u_c, I') \in Safe_{(9)} \subseteq \to_{(9)} \text{ if } \forall x \in I, x' \in I' \\ (x, u_c, x') \in \to_{(2)} \Rightarrow (x, u_c, x') \in Safe_{(2)}\end{gathered} \tag{12}$$

$$I_{m,(9)} := \{I \subseteq 2^X : \ell(I) = \{q_m\}\} \tag{13}$$

Theorem 4.2: Let H_{2^X}, $H_{2^{\tilde{Q}}}$, and R be defined as in Thm. 4.1 and suppose that G is an exact state reduction of system (2). Define the safety specification for $Obs(G')$ as $Safe_{Obs} \subseteq 2^{\tilde{Q}} \times U_c \times 2^{\tilde{Q}}$ by $(\iota, u_c, \iota') \in Safe_{Obs}$ if and only if for all $(I, u_c, I') \in \to_{(9)}$ such that $(\iota, I) \in R$ and $(\iota', I') \in R$, we have that $(I, u_c, I') \in Safe_{(9)}$. Let the set of marked states of $Obs(G')$ be the singleton $\iota_{m,Obs} := \{\{q_m\}\}$. Let the maximally permissive, safe, and nonblocking supervisor of $Obs(G')$ with respect to $Safe_{Obs}$ be S. Define the supervisor $\sigma : 2^X/\ell \to 2^{V_c}$ by $\sigma(I) = \{u_c/\tau : u_c \in S(\ell(I))\}$. Then σ solves Prob. 4.1.

V. Conclusion

We considered the problem of collision avoidance in vehicular networks in the presence of uncontrolled vehicles, a disturbance, and imperfect measurements. Specifically, given a system of vehicles crossing an intersection, we sought to obtain a maximally permissive supervisor that ensured that all vehicles cross the intersection safely, despite imprecisely measuring vehicle positions. We defined the concept of a state reduction and proved that, when one system is a state reduction of the other, we may obtain a supervisor that is safe, non-blocking, and maximally permissive for the system with the larger state space by translating the safety and non-blocking specifications to the reduced system, using standard supervisory control techniques of DES to solve for the maximally permissive, safe, and non-blocking supervisor for the reduced system, and then translating the resulting supervisor back to the original system. We constructed a new discrete event system abstraction by introducing a finite set of observable but uncontrollable "measurement events", showed that this abstraction was a state reduction of a system defined over state estimates, and used this abstraction to obtain the desired supervisor for the continuous domain system.

Future work will proceed in three directions: generalizing the definition of state reduction to deal specifically with partially observable automata in order to extend the main result of Thm. 4.2 to the case where G is an inexact state reduction of system (2); finding computationally efficient algorithms for our solution method, as was done in [7]; and extending our results to the case of a second order system.

References

[1] T.-C. Au, C.-L. Fok, S. Vishwanath, C. Julien, and P. Stone, "Evasion planning for autonomous vehicles at intersections," in *IEEE/RSJ International conference on Intelligent Robots and Systems*, 2012.

[2] C. G. Cassandras and S. Lafortune, *Introduction to Discrete Event Systems*. Springer-Verlag, 2008.

[3] A. Colombo and D. Del Vecchio, "Efficient algorithms for collision avoidance at intersections," in *Hybrid Systems: Computation and Control*, 2012.

[4] ——, "Enforcing safety of cyber-physical systems using flatness and abstraction," in *Proceedings of the Work-in-Progress session of ICCPS 2011*, 2011.

[5] ——, "Supervisory control of differentially flat systems based on abstraction," in *50th IEEE Conference on Decision and Control*, 2011.

[6] A. Colombo and A. Girard, "An approximate abstraction approach to safety control of differentially flat systems," in *European Control Conference*, 2013.

[7] E. Dallal, A. Colombo., D. Del Vecchio, and S. Lafortune, "Supervisory control for collision avoidance in vehicular networks using discrete event abstractions," in *American Control Conference*, 2013.

[8] M. Hafner and D. Del Vecchio, "Computation of safety control for uncertain piecewise continuous systems on a partial order," in *48th IEEE Conference on Decision and Control*, pp. 1671–1677, 2013.

[9] H. Kowshik, D. Caveney, and P. R. Kumar, "Provable systemwide safety in intelligent intersections," *IEEE Trans. Veh. Technol.*, vol. 60, pp. 804–818, 2011.

[10] S. M. LaValle, *Planning algorithms*. Cambridge university press, 2006.

[11] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event processes," *SIAM J. Control and Optimization*, vol. 25, no. 1, pp. 206–230, Jan. 1987.

[12] P. Tabuada, *Verification and control of hybrid systems*. Springer-Verlag, 2009.

[13] C. J. Tomlin, I. Mitchell, A. M. Bayen, and M. Oishi, "Computational techniques for the verification of hybrid systems," *Proc. IEEE*, vol. 91, pp. 986–1001, 2003.

[14] R. Verma and D. Del Vecchio, "Semiautonomous multivehicle safety: A hybrid control approach," *IEEE Robotics & Automation Magazine*, vol. 18, pp. 44–54, 2011.

[15] W. M. Wonham and P. J. Ramadge, "On the supremal controllable sublanguage of a given language," *SIAM J. Control and Optimization*, vol. 25, no. 3, pp. 637–659, May 1987.