On the relationship between bisimulation and combinatorial filter reduction

Hazhar Rahmani and Jason M. O'Kane

***Abstract*— Combinatorial filters are discrete structures for modeling and reasoning about robotic systems. Such filters are of interest not only because of the potential for reduction of the computational power needed to execute the filter, but also for the insight they can sometimes provide into the information requirements of certain robotic tasks.**

It is known that the filter minimization problem —that is, for a given filter, to find a combinatorial filter with the minimal number of states among all filters with equivalent behavior— is NP-hard. Intuition might suggest that the well-known notion of *bisimulation* might be of direct use for this minimization problem. Indeed, the bisimilarity relation —the union of all bisimulation relations over the state space of the original filter— is an equivalence relation, and one might attempt to reduce a filter by merging states that are equivalent under this relation.

This paper studies this relationship between bisimulation and combinatorial filter reduction. Specifically, we show that every filter minimization problem can be solved by computing a quotient of the input filter with some relation, but that for some filters, the bisimilarity relation is *not* the correct relation for this purpose. We also characterize the result of the bisimulation quotient operation as the solution to a different, stricter filter minimization problem, and identify several classes of filters for which a variant of bisimulation, called compatibility, can be used to minimize filters in polynomial time.

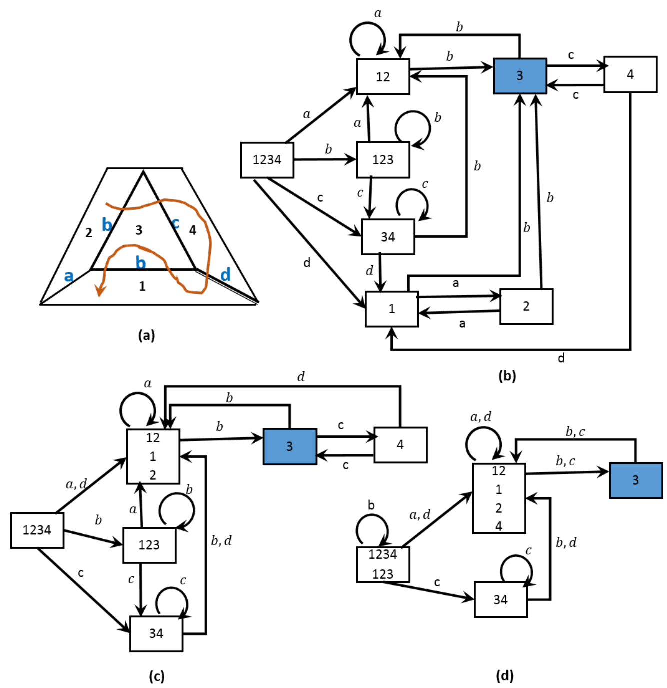

Fig. 1. **(a)** An agent moves in an environment divided into four regions by five beam sensors. **(b)** A filter for that system that provably tells if the agent is region 3 or not. **(c)** A filter that mimics the behavior of the original filter, obtained by bisimulation-quotienting. **(d)** The smallest filter equivalent to the original filter. Note in particular that reduction via bisimulation does not produce a fully-minimized filter.

I. Introduction

Combinatorial filters, first proposed by LaValle [12], [13], are a general class of models for reasoning about systems that process discrete (rather than continuous) sensor data. For these kinds of filters, a natural and important question is to consider the number of states required to express the desired behavior as labelled transition graph. This problem of reduction of combinatorial filters was addressed by O'Kane and Shell [16], who proved that this problem is NP-hard.

Similar problems relating to the minimization of discrete transition structures have been studied through the lens of bisimulation relations [25]. Informally, a bisimulation relation over the states of a transition system includes pairs of states that can be 'merged' without impacting the behavior of the system. In that context, a relatively straightforward minimization algorithm, which we call *bisimulation-quotienting*, would find the largest bisimulation relation, which is known to be a unique equivalence relation, and then to merge equivalent states into a single state. Can this bisimulation-based approach be used for reduction of combinatorial filters?

This paper provides an answer to that question in the negative, by exploring the relationship between bisimulation relations and combinatorial filter reduction. Figure 1 illustrates an example, inspired by the work of Tovar *et al.* [27]. Figure 1a shows an environment, divided into four regions by five beam sensors, in which an agent moves. When the agent crosses a beam, the system can tell the label of crossed beam —a, b, c, or d— but not the direction of the crossing. In addition, the agent can travel from region 4 to region 1 only in that direction; direct travel from region 1 to region 4 is impossible. Note, for example, that from region 3, crossing of a b beam can be triggered by the agent moving to region 1 or region 2; the system cannot directly distinguish these two possibilities. The task of the system is to determine, at any time, whether the agent is definitely in region 3.

A naïve combinatorial filter for this task appears in Figure 1b. Each of the eight states in this filter corresponds to a set of possible regions that might contain the agent. Edges indicate changes to that set that result from each beam crossing that might be observed. Colors on the states indicate the filter's output, with the darker node corresponding to certainty that the agent is in region 3 and the lighter

The authors are with the Department of Computer Science and Engineering, University of South Carolina, Columbia, South Carolina, USA. hrahmani@email.sc.edu, jokane@cse.sc.edu
This material is based upon work supported by the National Science Foundation under Grant No. 1526862.

nodes indicating otherwise. Figure 1c depicts the result of bisimulation-quotienting on this original filter. In this case, bisimulation-quotienting reduced the size of the original filter, but not optimally so, as shown by the actual optimal filter in Figure 1d.

The broader question we address here is to find an equivalence relation, similar in spirit to the bisimilarity relation on the state space of the original filter, that when used for constructing a quotient filter actually does lead to an optimally reduced filter. Saberifar *et al.* [24] proved that such an equivalence relation does indeed exist. However, their proof is not constructive, and no readily-constructed equivalence relation over the states in the original filter for this purpose is known. This paper clarifies this situation by introducing a variant of bisimulation called *compatibility* and showing that the equivalence relation that leads to optimal filter reduction is a maximal subset of the union of all compatibility relations.

After reviewing related work in Section II and recalling the basic definitions in Section III, this paper presents several new contributions.

- In Section IV, we provide basic ideas to consider solving the filter minimization problem as making a quotient filter under a certain kind of relation, which we call *compatibility relation*. We prove that the state space of any minimal filter is the quotient of the state space of the original filter under a compatibility equivalence relation.
- In Section V, we show why the use of bisimilarity relation in making the quotient filter fails to produce an optimal solution for the filter minimization problem. The intuition is that, for bisimulation-quotienting, the set of observation sequences that can be processed by the reduced filter, called the *language of the filter*, must be identical to the language of the original filter; in contrast, for the combinatorial filter reduction problem, the language of the reduced filter may be a superset of the language of the original filter.
- In Section VI, we show that bisimilarity-quotienting induces a filter with the smallest number of states among all filters who behave the same as the original filter, and whose languages are identical to the language of the original filter.
- In Section VII, we show that the union of all compatibility relations is not in general an equivalence relation, and thus, cannot be always used to making a quotient filter. But, if it is an equivalence relation for a given filter, then the quotient filter under the union of all compatibility relations is indeed a minimal filter for the given filter. We use this idea to identify several classes of filters for which filter minimization problem can be solved efficiently.

Concluding remarks appear in Section VIII.

II. Related Work

Building on a foundation of prior work on minimalism in robotics [4], [7], combinatorial filters were originally formulated by LaValle [12], [13]. The key idea is to make, from the data accessible to the robot, a smallest abstraction still adequate to solve a given task. This approach has recently been utilized for a wide spectrum of tasks including navigation [14], [26], [28], exploration [11], manipulation [10], target tracking [2], [32], and story validation [31].

Interest in forming combinatorial filters that are minimal, in the sense of minimizing the number of states, is motivated not only by the reduction in resources needed to execute such filters, but also by the insight into the nature of the underlying problems that arises from identifying the information required to solve those problems. The problem of performing this reduction automatically was first studied by O'Kane and Shell [16], who proved via a reduction from the graph 3-coloring problem that the filter minimization problem is NP-hard. Saberifar *et al.* [23] showed that several special cases of filters, including tree and planar filters, remain hard to minimize, and that the filter minimization problem is hard even to approximate with ϵ-guarantee, for any ϵ.

Bisimulation was discovered independently in at least three different fields–in modal logic, by van Benthem [29]; in process theory, independently by Milner [15] and Park [19]; and in set theory, by Forti and Honsell [6]. It is currently used across many fields, including automata and language theory [21], [20], coalgebra and coinduction [5], [22], and dynamical and control systems [8], [30]. Generally speaking, bisimulation can be used for at least two purposes: either to prove that two objects are behaviorally equivalent, or to minimize the size of a structure by forming the quotient under the coarsest bisimulation equivalence relation between elements of the original structure. This paper focuses on the latter application. Computing this coarsest bisimulation equivalent relation is generally performed using partition refinement algorithms [9], [18]. Details about bisimulation quotient algorithms appear in the survey by Cleaveland and Sokolsky [3].

III. Definitions

This section presents basic definitions used throughout the paper. We are interested in filters that model the behavior of a robot in response to a discrete, finite sequence of observations. The following definitions are equivalent to the those introduced by O'Kane and Shell [17].

Definition 1: *A* filter *is a 6-tuple* $(V, Y, C, \delta, c, v_0)$ *in which:*

- V *is a finite set of* states*,*
- Y *is a set of possible* observations*, representing the input space of the filter,*
- C *is a set of* outputs*, sometimes called* colors*, representing the outputs produced by the filter,*
- $\delta : V \times Y \to V \cup \{\bot\}$ *is the* transition function *of filter,*
- $c : V \to C$ *is a function assigning to each state* $v \in V$ *a color, and*
- $v_0 \in V$ *is the initial state.*

Filters are readily shown as directed graphs, in which the states are vertices and edges are determined by the transition function. Recall the examples in Figure 1b–d.

For state-observation pairs (v, y) for which $\delta(v, y) = \perp$, we interpret this to mean that we can be sure that observation y will not, because of some structure in the robot's environment, occur when the filter is in state v. In the graph view, there would simply be no outgoing edge from v labeled y.

Note that Definition 1 ensures that from any state, for any observation, at most one transition can happen. The next definition makes this idea more precise.

Definition 2: *Let $F = (V, Y, C, \delta, c, v_0)$ be a filter, $v \in V$ be a state, and $s = s_1 s_2 ... s_n \in Y^*$ be an observation sequence where each s_i is a member of Y. We say that s is* trackable *from v if there is a sequence of states $q_0, q_1, ..., q_n$ such that:*

- *$q_0 = v$, and*
- *$\delta(q_i, s_{i+1}) = q_{i+1}$ for all $0 \leq i < n$.*

Given a state $v \in V$ and an observation sequence $s \in Y^*$ trackable from v, we write $\delta^*(v, s)$ to denote the state reached by tracing s starting from v. If s is not trackable from v, we write $\delta^*(v, s) = \perp$. For the empty string ϵ, we define $\delta^*(v, \epsilon) = v$ for all states v, and define that ϵ is trackable for all states v.

We can now define the language of a filter, which plays a crucial role in filter reduction.

Definition 3: *The* language of a state *v, denoted $L(v)$, is the set of all observation sequences trackable from v. The* language of a filter *F, denoted $L(F)$, is the language of its initial state: $L(F) = L(v_0)$.*

Before we can speak meaningfully about reduction of filters, we need a definition of filter equivalence with respect to a language.

Definition 4: *Let $F_1 = (V_1, Y, C, \delta_1, c_1, v_0)$ and $F_2 = (V_2, Y, C, \delta_2, c_2, w_0)$ be two filters with the same observation space Y and the same color space C. Let $L \subseteq Y^*$ denote a language of observation sequences. We say that F_1* is equivalent to *F_2* with respect to *L, denoted $F_1 \stackrel{L}{=} F_2$, if for any observation sequence $s \in L$:*

- *$\delta_1^*(v_0, s) \neq \perp$,*
- *$\delta_2^*(w_0, s) \neq \perp$, and*
- *$c_1(\delta_1^*(v_0, s)) = c_2(\delta_2^*(w_0, s))$.*

This definition says that any observation sequence that is in L is trackable by both F_1 and F_2, and both of them produce the same output while tracing that sequence. However, for observation sequences that are not in L, it does not say anything about the outputs generated by the two filters, nor does it require that the observation languages of F_1 and F_2 should be the same. The central problem this work studies is called *the filter minimization problem*.

Problem: Filter minimization (FM) [17]

Input: A filter F.

Output: A filter F^* such that $F \stackrel{L(F)}{=\!=} F^*$ and the number of states in F^* is minimal.

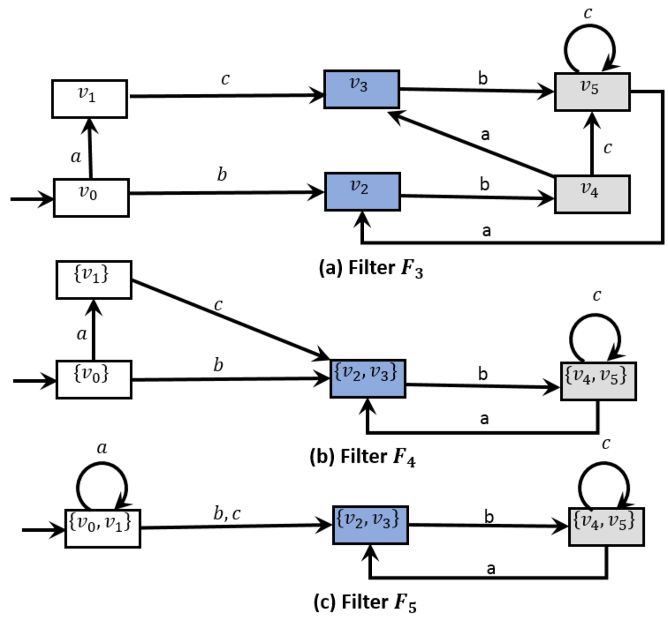

Fig. 2. **a)** A sample filter F_3. **b)** A minimal filter F_4 such that $F_3 \stackrel{L(F_3)}{=\!=} F_4$ and $L(F_3) = L(F_4)$. **c)** A minimal filter F_5 such that $F_3 \stackrel{L(F_3)}{=\!=} F_5$.

Note that we allow that $L(F) \subset L(F^*)$ because, for the observation sequences that are in $L(F^*) - L(F)$, we are sure that they will never occur due to the properties of the environment and the problem.

Because much of what follows deals with relations over the set of a filter's state, we will rely on some elements of standard notation for such relations. For a given filter F, we use I_F to denote the identity relation on the state set V of F, i.e. $I_F = \{(v, v) \mid v \in V\}$. If $R \subseteq V \times V$ is an equivalence relation on V, then it partitions V to a set of equivalence classes. For any $v \in V$, the equivalence class of v in R is denoted $[v]_R$, so that $[v]_R = \{w \in V \mid (v, w) \in R\}$. In particular, for any $v, w \in V$, if $(v, w) \in R$, then $[v]_R = [w]_R$. Finally, the set of all equivalence classes of R is called the quotient of V under R, denoted V/R.

IV. Filter reduction as a quotient operation

In this section, we show how the process of filter minimization can be understood as a quotient operation with respect to certain kinds of relations over the states of the input filter.

The intuition is that we want to consider relations that indicate which pairs of states should be 'merged' to form a reduced filter. Therefore, we must establish conditions on the relation that guarantee that this merging operation makes sense.

Definition 5: *Let $F = (V, Y, C, \delta, c, v)$ be a filter and let $R \subseteq V \times V$ denote a relation over the states of F. We say that R is a* compatibility relation *for F, if for any $(v, w) \in R$:*

1) *$c(v) = c(w)$, and*
2) *for any $y \in Y$, if $\delta(v, y) \neq \perp$ and $\delta(w, y) \neq \perp$, then $(\delta(v, y), \delta(w, y)) \in R$.*

To illustrate this definition, consider filter F_3, depicted in Figure 2. Some compatibility relations for F_3 are $R_1 = \emptyset$, $R_2 = \{(v_0, v_1)\}$, and $R_3 = \{(v_3, v_2), (v_5, v_4), (v_5, v_5), (v_2, v_3), (v_4, v_5)\}$.[1]

Now we can define the notion of a quotient filter.

Definition 6: *For a filter $F = (V, Y, C, \delta, c, v_0)$, and a relation $R \subseteq V \times V$ that is both a compatibility relation and an equivalence relation (a compatibility equivalence relation), the* quotient of F under R *is the filter $F/R = (V/R, Y, C, \delta', c', [v_0]_R)$, in which*

$$\delta'([v]_R, y) = \begin{cases} [\delta(w, y)]_R & \text{if } \exists w \in [v]_R \text{ with } \delta(w, y) \neq \bot \\ \bot & \text{otherwise} \end{cases}$$

and $c'([v]_R) = c(v)$.

Note that Definition 5 ensures that every transition in a quotient filter is well-defined. Because R must be a compatibility relation, if two states v and w that share some outgoing observation y are merged, then the resulting states $\delta(v, y)$ and $\delta(w, y)$ must be merged as well. Consider filter F_3 depicted in Figure 2. A compatibility equivalence relation for this filter is $R = I_{F_3} \cup \{(v_3, v_2), (v_5, v_4), (v_2, v_3), (v_4, v_5)\}$. The quotient of F_3 under this relation, F_3/R, is filter F_4, depicted in Figure 2.

The next two lemmas establish that, though this quotient operation may increase the language of the filter, it does not change the behavior, in the sense of Definition 4.

Lemma 1: For any filter $F = (V, Y, C, \delta, c, v_0)$, and any compatibility equivalence relation R for F, $L(F) \subseteq L(F/R)$.

Lemma 2: For any filter $F = (V, Y, C, \delta, c, v_0)$, and any compatibility equivalence relation R for F, $F \xlongequal{L(F)} F/R$.

The proofs, which we omit for space reasons, proceed by induction on the length of the observation sequences, leveraging Definitions 4 and 6.

Next we prove that the state space of any optimally reduced filter can be seen as the quotient of the state space of the original filter under some compatibility equivalence relation. We begin with a result due to Saberifar *et al.*:

Lemma 3: [24] Let $F_1 = (V_1, Y, C, \delta_1, c_1, v_0)$ denote a filter that does not have any unreachable states, and let $F_2 = (V_2, Y, C, \delta_2, c_2, w_0)$ denote some minimal filter for which $F_1 \xlongequal{L(F_1)} F_2$. Then there exists a function $f : V_1 \to V_2$ such that for every observation sequence $s \in L(F_1)$, $\delta_2^*(w_0, s) = f(\delta_1^*(v_0, s))$.

We now strengthen this result slightly.

Lemma 4: The function described in Lemma 3 is surjective.

[1]Note that the notion of compatibility relation is different from the usual notion of simulation, in that its second condition in weaker is required of a simulation relation. In fact, two states can be compatible (can exist in a compatibility relation) while none of them simulates another. As an example, states 2 and 4 in Figure 1b are compatible but neither of them simulates the other.

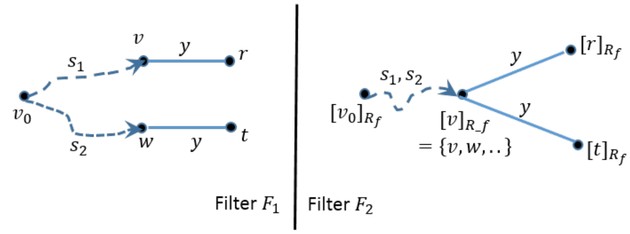

Fig. 3. An illustration of the proof of Lemma 5. Filter F_2 is a minimal filter for which $F_1 \xlongequal{L(F_1)} F_2$ holds. The state space of F_2 corresponds to the quotient of the state space of F_1 under the equivalence relation R_f. The assumption for this relation is $(v, w) \in R_f$ but $(r, t) \notin R_f$.

Proof: We prove that each state in F_2 is mapped to by at least one state in F_1 via f. For the sake of contradiction, suppose that there exists a state z in F_2 that is not mapped to by f from any state in F_1. We consider two cases.

1) If no observation sequence that ends or passes through z is in $L(F_1)$, then we can construct a new filter F_3 from F_2 by removing state z. Clearly, $F_1 \xlongequal{L(F_1)} F_3$ and F_3 has fewer states than F_2. This contradicts the construction that F_2 is minimal.
2) If there exists an observation sequence $s \in L(F_1)$ that ends or passes through z when traced in F_2, then let $k \leq |s|$ be an integer such that $\delta_2^*(w_0, s_{1...k}) = z$. By the structure of filters, and given that $s \in L(F_1)$, we conclude that $s_{1...|s|-1} \in L(F_1)$, $s_{1...|s|-2} \in L(F_1)$, $\ldots$, and ultimately $s_{1...k} \in L(F_1)$. This by Lemma 3 proves that z is mapped to by $\delta_1^*(v_0, s_{1...k})$, which is a contradiction.

■

Given such a function f, we define an equivalence relation $R_f \subseteq V \times V$ so that $(v, w) \in R_f$ if and only if $f(v) = f(w)$. Note that there is a one-to-one correspondence between the equivalence classes $[v]_{R_f}$ of R_f and the states of F_2.

Lemma 5: For any filter F_1 and any minimal equivalent filter F_2, the equivalence relation R_f they induce is a compatibility relation.

Proof: First observe that by the construction of R_f, for any $v, w \in V$, if $(v, w) \in R_f$, then v and w are mapped to a single state in F_2. Let $[v]_{R_f}$ be such a state. To show that R_f is a compatibility relation, we prove that conditions (1) and (2) of Definition 5 hold for any v and w for which $(v, w) \in R_f$. Suppose that condition (1) does not hold, that is, $c_1(v) \neq c_1(w)$, which means $c_2([v]_R)$ is different from $c_1(v)$ or $c_1(w)$. Without loss of generality assume that $c_1(v) \neq c_2([v]_{R_f})$. Let $s \in L(F_1)$ by an observation sequence such that $\delta_1^*(v_0, s) = v$. By Lemma 3, we have that $\delta_2^*(w_0, s) = [v]_{R_f}$. But, $c_1(\delta_1^*(v_0, s)) = c_1(v) \neq c_2([v]_{R_f}) = c_2(\delta_2^*(w_0, s))$, which by Definition 4 contradicts that $F_1 \xlongequal{L(F_1)} F_3$.

Now suppose that condition (2) does not hold, which means that there exists $y \in Y$, such that $\delta_1(v, y) \neq \bot$ and $\delta_1(w, y) \neq \bot$ but $(\delta_1(v, y), \delta_1(w, z)) \notin R_f$. Let $r = \delta_1(v, y)$ and $t = \delta_1(w, y)$. We argue that if this is the case, then F_2 is not a filter, which is a contradiction. Figure 3 illustrates this proof. Let s_1 and s_2 be two observation sequences that end in v and w, respectively, when traced by F_1. By Lemma 3, states v and w are in the same equivalence class of R_f, and thus, they are mapped to a single state, such as $[v]_{R_f}$, in F_2; hence, both s_1 and s_2 end in $[v]_{R_f}$ when traced by F_2. Consider also that observation sequences s_1y and s_2y end in r and t, respectively, when traced by F_1. Observe that from state $[v]_{R_f}$ there should be two outgoing edges with the same label y, one of which goes to $[r]_{R_f}$ and another goes to $[t]_{R_f}$. Because F_2 is a filter, the only way to reach r by tracing s_1y from the initial state is to have an edge labeled by y, that goes from $[v]_{R_f}$ to $[r]_{R_f}$. We can use the same argument to prove that there should be an outgoing edge labeled by y that connects $[v]_{R_f}$ to $[t]_{R_f}$. This implies that F_2 has two edges labeled y from $[v]_{R_f}$, a contradiction. ■

In particular, since R_f is both an equivalence relation and a compatibility relation for F_1, it is meaningful to consider the quotient filter F_1/R_f. Moreover, F_1/R_f is structurally identical to the minimal filter F_2. Of course, in the context of filter minimization, F_2 is unknown, so we cannot expect to compute F_1/R_f directly. However, the impact of Lemma 5 is that we can view the problem of filter minimization as equivalent to the problem of identifying a suitable compatibility equivalence relation with which to construct a quotient filter —there always exists some such relation for which the quotient leads to the minimal filter. The question remains, however: Which compatibility equivalence relation is the right one?

V. Bisimulation and filter reduction

One apparently reasonable hypothesis is that the notion of bisimulation may be useful for filter minimization via compatibility equivalence relation quotient. This section explores that idea, and shows that although the bisimilarity relation is indeed a compatibility equivalence relation, it does not in general induce minimal filters. We begin by adapting the standard notion of bisimulation within states of a transition system to filters.

Definition 7: *Let $F = (V, Y, C, \delta, c, v_0)$ be a filter. A relation $R \subseteq V \times V$ is said to be a* bisimulation relation *for F if for any $(v, w) \in R$:*

1) *$c(v) = c(w)$,*
2) *for any $y \in Y$, if $\delta(v, y) \neq \bot$, then $\delta(w, y) \neq \bot$ and $(\delta(v, y), \delta(w, y)) \in R$*
3) *for any $y \in Y$, if $\delta(w, y) \neq \bot$, then $\delta(v, y) \neq \bot$ and $(\delta(v, y), \delta(w, y)) \in R$*

We say that state v in filter F is *bisimilar* to state w in filter F if there exists a bisimulation relation R for F such that $(v, w) \in R$.

Observe that any union of bisimulation relations for a filter is itself a bisimulation relation. The union of *all* bisimulation relations for F, denoted $\sim_F$, is called the *bisimilarity relation* for F. Recall filter F_3, depicted in Figure 2. For this filter, we have $\sim_{F_3} = I_{F_3} \cup \{(v_2, v_3), (v_3, v_2), (v_4, v_5), (v_5, v_4)\}$.

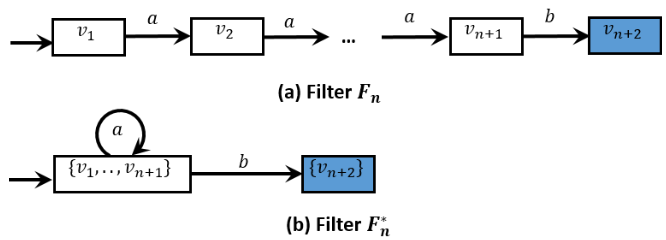

Fig. 4. a) It shows the construction of filter F_n, mentioned in Theorem 1. The quotient of this filter under $\sim_{F_n}$ does not reduce its size. b) Filter F_n^* is the minimal for F_n with respect to $L(F_n)$. State v_{n+2} in filter F_n and state $\{v_{n+2}\}$ in filter F_n^* have color 2; all other states in both filters have color 1.

Such bisimilarity relations are of interest in part because they are suitable for constructing quotient filters.

Lemma 6: The bisimilarity relation of every filter is both a compatibility relation and an equivalence relation.

Proof: It is easy to prove that the bisimilarity is an equivalence relation (for a proof, see Lemma 7.8 of [1]). Also, by Definition 7, any bisimulation relation —including the bisimilarity relation— is a compatibility relation in the sense of Definition 5. ■

Because the bisimilarity relation of a given filter represents, in a certain sense, a coarsest partitioning of the states into ‘mergable’ subsets, intuition might suggest that a quotient with the bisimilarity relation might perhaps produce an optimally reduced filter, in the sense of the FM problem. The next result debunks this misconception.

Theorem 1: For any integer $n \geq 1$, there exists a filter F_n with $n + 2$ states, such that $F_n/\sim_{F_n}$ is larger than the optimal solution F_n^* to the filter minimization problem FM by n states.

Proof: For a given n, we construct a filter F_n with $n+2$ states, for which $F_n/\sim_{F_n}$ also has $n + 2$ states. Figure 4a shows the construction. In particular, note that for any pair of distinct states (v, w), we have $v \nsim_{F_n} w$; this is because if $v \sim_{F_n} w$, then they must have the same color, meaning that there must exist $1 \leq i \neq j \leq n + 1$ such that $v = v_i$ and $w = v_j$, and if this is the case then by the definition of bisimulation relation, we must have that $v_{i+1} \sim_{F_n} v_{j+1}$, $v_{i+2} \sim_{F_n} v_{j+2}$, ..., and ultimately $v_{i+k} \sim_{F_n} v_{n+2}$, which is a contradiction. Therefore $\sim_{F_n} = I_{F_n}$, and $F_n/\sim_{F_n}$ is structurally identical to F_n — no two states will be merged. In contrast, for any n, the optimally reduced filter F_n^* has exactly two states, as shown in Figure 4b. ■

In particular, Theorem 1 implies that bisimulation-quotienting does not always induce an optimal solution to the filter minimization problem FM, and in fact, that the difference in size between the optimally reduced filter and the bisimilarity-quotient filter cannot be bounded.

VI. STRONG FILTER MINIMIZATION

Section V showed that, although quotient with bisimilarity relation produces an equivalent filter, that filter may not necessarily be minimal. In this section, we provide some insight into why that happens, by showing that this kind of bisimilarity quotient instead solves a variant of the filter minimization problem, in which the language of the reduced filter must be identical to the language of the original filter, rather than merely a superset of it. Specifically, this section shows that bisimilarity-quotienting solves the following problem.

Problem: Strong Filter Minimization (SFM)

Input: A filter F.

Output: A filter F^* such that $F \xlongequal{L(F)} F^*$, $L(F) = L(F^*)$, and the number of states in F^* is minimal.

Now we can state the main result of this section.

Theorem 2: For any filter F, the bisimilarity quotient $F/\sim_F$ is a solution to the SFM problem for F.

The proof, again omitted due to space limitations, uses the same techniques as the proof of the very similar well-known result for finite labeled transition systems [1], [15]; the primary difference is that, for filters, we are concerned with finite-length, rather than infinite, input sequences.

Corollary 1: SFM can be solved in polynomial time.

Proof: Beyond Theorem 2, we need only to show that given a filter $F = (V, Y, C, \delta, c, v_0)$ both (a) the bisimilarity relation $\sim_F$ and (b) the quotient of a filter and a relation, can be computed in polynomial time.

A simple efficient algorithm for constructing the bisimilarity relation starts with assigning the set $\{(v, w) \in V \times V \mid c(v) = c(w) \wedge \forall y \in Y, (\delta(v, y) = \delta(w, y) = \perp \vee c(\delta(v, y)) = c(\delta(w, y)))\}$ as the initial value to a variable R. Then, in each iteration of a loop, all members of R that fail to satisfy all three conditions of Definition 7 are removed from R. This loop continues until no additional members of R can be removed; at that time, we have $R = \sim_F$. Clearly, the time complexity of this algorithm is $O(|V|^4 \times |Y|)$. This relation has at most $|V|^2$ members, hence, the filter $F/\sim_F$ is constructed in $O(|V|^4 \times |Y|)$ time. ■

As an example of this theorem, consider again filter F_3 from in Figure 2. The quotient of this filter under $\sim_{F_3}$ is filter F_4, depicted in the same figure The language of F_2 is equal to the language of F_3. Filter F_5, depicted in the same figure, represent the smallest filter who is equivalent to F_3 with respect to the language of F_3. In this case, we have $L(F_3) \subset L(F_5)$.

Knowing now that making the quotient of a filter under bisimilarity relation does not always optimally reduce the size of that filter, in the next section, we are interested in other compatibility relations that are better suited for filter reduction.

VII. SPECIAL CLASSES OF FILTERS

By the discussions of the Section IV, to minimize a given filter F —one without any unreachable states, of course— one can make the quotient filter under some compatibility equivalence relation. Section V proved that the bisimilarity relation $\sim_F$ is not always the appropriate relation for this job.

Another intuitive possibility would be to use the union of all compatibility relations, analogous to the definition of the bisimilarity relation as the union of all bisimulation relations. As an example, this relation for filter F_3 depicted in Figure 2, is $I_{F_3} \cup \{(v_0, v_1), (v_1, v_0), (v_2, v_3), (v_3, v_2), (v_4, v_5), (v_5, v_4)\}$. For a given filter F, we write $\curlywedge_F$ to denote this the union of all compatibility relations for F. Trivially, $\curlywedge_F$ is itself a compatibility relation for F.

In addition, we can compute $\curlywedge_F$ in time polynomial in the size of F. See Algorithm 1 for a simple approach to doing so. The intuition is to begin with a relation containing state pairs that are compatible for observation strings of length at most one, and then to iteratively delete state pairs that violate Definition 5 for successively longer strings. The time complexity of this algorithm is $O(|V|^4|Y|)$, where V and Y are, respectively, the state space and the observation space of the input filter.

The next lemma shows that, unfortunately, $\curlywedge_F$ may not be suitable for forming quotient filters, because for some filters it is not an equivalence relation. (Recall Definition 6, under which quotient filters are well-defined only for compatibility equivalence relations.)

Lemma 7: For any filter $F = (V, Y, C, \delta, c, v)$, the relation $\curlywedge_F$ is reflexive and symmetric. However, there exist filters F for which $\curlywedge_F$ is not transitive.

Proof: For the first claim, consider that in sense of Definition 5, the identity relation $I_F = \{(v, v) \mid v \in V\}$ is a compatibility relation for F. By definition of $\curlywedge_F$, it is a superset of I_F, and therefore $\curlywedge_F$ is reflexive. To prove that $\curlywedge_F$ is symmetric, one need to show that if $v \curlywedge_F w$, then $w \curlywedge_F v$. Suppose that $v \curlywedge_F w$. This means that there exists a compatibility relation R for F such that $(v, w) \in R$. By the symmetry of conditions (1) and (2) of Definition 5 with respect to v and w, if R is a compatibility relation for F, then so is R^{-1}. The relation R^{-1} contains (w, v), and so does $\curlywedge_F$ given the definition of $\curlywedge_F$.

For the second claim, to observe that $\curlywedge_F$ may not be transitive, let F be the filter depicted in Figure 5. For this filter, we have $\curlywedge_F = I_F \cup \{(v_1, v_2), (v_2, v_1), (v_2, v_3), (v_3, v_2)\}$. This relation is not transitive since $v_1 \curlywedge_F v_2$ and $v_2 \curlywedge_F v_3$ hold but $v_1 \curlywedge_F v_3$ does not. ■

In an important sense, Lemma 7 should not be a surprise. Since the filter minimization problem is NP-hard [17], [24], and $F/\curlywedge_F$ can be computed in polynomial time, if Lemma 7 were false, that would imply that $P = NP$.

However, any filter F for which $\curlywedge_F$ is indeed an equivalence relation, then $F/\curlywedge_F$ is guaranteed to be the minimal filter equivalent to F. Moreover, given any filter F, it takes

Algorithm 1: UNIONOFALLCOMPRELATIONS

```
Input: A filter F = (V, Y, C, δ, c, v_0)
R ← ∅
forall (v, w) ∈ V × V do
    if c(v) ≠ c(w) then
        continue
    add ← true
    forall y ∈ Y do
        if δ(v, y) ≠ ⊥ and δ(w, y) ≠ ⊥ then
            if c(δ(v, y)) ≠ c(δ(w, y) then
                add ← false
    if add = true then
        R ← R ∪ {(v, w)}
updated ← true
while updated = true do
    updated ← false
    forall (v, w) ∈ R do
        forall y ∈ Y do
            if δ(v, y) ≠ ⊥ and δ(w, y) ≠ ⊥ then
                if (δ(v, y), δ(w, y)) ∉ R then
                    R = R/{(v, w)}
                    updatedd = true
return R
```

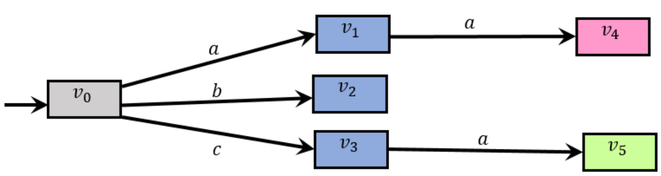

Fig. 5. A filter for which the union of all compatibility relations is not an equivalence relation. State v_0 has color 1, states in the middle column have color 2, state v_4 has color 3, and state v_5 has color 4.

polynomial time to check whether $\curlywedge_F$ is an equivalence relation or not. This implies that solving filter minimization problem for any filter for which $\curlywedge_F$ is an equivalence relation takes polynomial time in size of F. This fact gives a roadmap to recognize some classes of filters for which the filter minimization problem is in P, specifically by looking for classes of filters for which the union of compatibility relations can be proven to be an equivalence relation.

One such special class of filters consists of filters is one we call *observation-at-most-once-in-a-color* filters. An observation-at-most-once-in-a-color filter is a filter in which for any observation, from all states with the same color, there is at most one outgoing edge labeled by that observation. Such filters are a generalization of the class that Saberifar *et al.* [24] called *once-appearing-observations* filters. The difference between once-appearing-observations and observation-at-most-once-in-a-color filter is that in the former each observation appears only once while in the latter an observation can appear more than one time in the filter, but only once from the states of each color. The following theorem proves that solving filter minimization problem for this class takes polynomial time in size of the input filter.

> **Problem: Observation-at-most-once-in-a-color Filter minimization (OBS-AT-MOST-ONCE-IN-A-COL-FM)**
>
> *Input:* An observation-at-most-once-in-a-color filter F.
>
> *Output:* A filter F^* such that $F \xlongequal{L(F)} F^*$, and the number of states in F^* is minimal.

Theorem 3: OBS-AT-MOST-ONCE-IN-A-COL-FM $\in P$.

Proof: According to the discussion above, we need only prove that for any observation-at-most-once-in-a-color filter $F = (V, Y, C, \delta, c, v_0)$, the relation $\curlywedge_F$ is an equivalence relation. It is easy to observe that since in F no distinct states with the same color shares an outgoing edge labeled with the same observation, we have $\curlywedge_F = \{(v, w) \mid c(v) = c(w)\}$. This relation is clearly an equivalence relation, ∎

Another class consists of filters which we call *largest-compatibility-is-bisimilarity*— a filter for which the union of all compatibility relations coincides with the bisimilarity relation.

> **Problem: Largest-compatibility-is-bisimilarity Filter minimization (LAR-COMP-IS-BISIM-FM)**
>
> *Input:* A largest-compatibility-is-bisimilarity filter F.
>
> *Output:* A filter F^* such that $F \xlongequal{L(F)} F^*$, and the number of states in F^* is minimal.

Theorem 4: LAR-COMP-IS-BISIM-FM $\in P$.

Proof: For any filter F in this class, $\curlywedge_F = \sim_F$. By Lemma 6, the relation $\sim_F$ is an equivalence relation. ∎

A subclass of largest-compatibility-is-bisimilarity filters are filters Saberifar *et al.* [24] called *no-missing-edges*– filters for which, from any state, for any observation, there is an outgoing edge labeled by that observation. This kind of filters can be generalized to a class we call *color-no-missing-edges* filters. A filter is color-no-missing-edges if for any two states v and w for which $c(v) = c(w)$, for any observation y, if $\delta(v, y) \neq \bot$ then $\delta(w, y) \neq \bot$.

> **Problem: Color-no-missing-edges Filter minimization (COL-NO-MIS-EDG-FM)**
>
> *Input:* A color-no-missing-edges filter F.
>
> *Output:* A filter F^* such that $F \xlongequal{L(F)} F^*$, and the number of states in F^* is minimal.

Theorem 5: COL-NO-MIS-EDG-FM $\in P$.

Proof: By the definition of color-no-missing-edges for each observation y, and for any two states v and w that share the same color, we have that either $\delta(v, y) = \delta(w, y) = \bot$ or $(\delta(v, y) \neq \bot) \wedge (\delta(w, y) \neq \bot)$. In this case, the three conditions of Definition 7, taking $F_1 = F_2 = F$, are identical to the conditions of Definition 5. Therefore, $\curlywedge_F = \sim_F$, which by Lemma 6 is an equivalence relation.

∎

In spite of this good news for these (admittedly narrow) classes of filters, in general the union of all compatibility relations for a filter F is not always an equivalence relation. In those cases, we must instead seek a *coarsest compatibility equivalence relation* —that is, a compatibility equivalence relation whose number of equivalence classes is smallest—to construct a minimal filter.

Theorem 6: Let F be a filter and let R be a coarsest compatibility equivalence relation. The filter F/R is a minimal filter for which $F \stackrel{L(F)}{=\!=} F^*$ holds.

Proof: By Lemma 2 and that R is a compatibility equivalence relation, $F \stackrel{L(F)}{=\!=} F^*$. If F/R is not minimal, that is if there is another filter F_2 with fewer number of states than F/R, then by Lemma 5, the relation R would not be a coarsest compatibility equivalence relation. ■

VIII. Conclusion

In this paper, we showed that the bisimulation quotient, which is widely used for reducing the size of transition systems, is not always appropriate for optimally reducing the size of combinatorial filters. However, we also showed that it is useful when one needs to prevent expansion of the language of a filter under minimization. We conclude that filter minimization can be done by making the quotient filter under a coarsest compatibility equivalence relation. In particular, if the union of all compatibility relations for a filter is an equivalence relation, then one can optimally reduce the size of that filter. By way of example, we identified several classes of filters for which this is the case.

Knowing that making the quotient of a filter under a coarsest compatibility equivalence relation optimally reduces the size of the filter, we believe that future work should consider the design of efficient heuristic algorithms for finding a coarsest compatibility equivalence relation. It is also interesting to attempt to identify practical filters for which finding the coarsest compatibility equivalence relation can be done in polynomial time.

Finally, there are some kinds of filters for which finding a coarsest compatibility equivalence relation can be directly solved by finding a minimum clique partitioning of the union of all compatibility relations. Several classes of graphs for which clique partitioning are in P have been recognized. This approach may provide a roadmap for finding additional classes of filters that can be minimized in polynomial time.

References

[1] C. Baier, J.-P. Katoen, and K. G. Larsen, *Principles of model checking*. MIT press, 2008.

[2] L. Bobadilla, O. Sanchez, J. Czarnowski, and S. M. LaValle, "Minimalist multiple target tracking using directional sensor beams," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2011.

[3] R. Cleaveland and O. Sokolsky, "Equivalence and preorder checking for finite-state systems," in *Handbook of Process Algebra*, 12 2001, pp. 391–424.

[4] M. A. Erdmann and M. T. Mason, "An exploration of sensorless manipulation," *IEEE Journal on Robotics and Automation*, vol. 4, no. 4, pp. 369–379, 1988.

[5] M. P. Fiore, "A coinduction principle for recursive data types based on bisimulation," *Information and Computation*, vol. 127, no. 2, pp. 186–198, 1996.

[6] M. Forti and F. Honsell, "Set theory with free construction principles," *Annali della Scuola Normale Superiore di Pisa-Classe di Scienze*, vol. 10, no. 3, pp. 493–522, 1983.

[7] K. Y. Goldberg, "Orienting polygonal parts without sensors," *Algorithmica*, vol. 10, no. 2, pp. 201–225, 1993.

[8] E. Haghverdi, P. Tabuada, and G. J. Pappas, "Bisimulation relations for dynamical, control, and hybrid systems," *Theoretical Computer Science*, vol. 342, no. 2-3, pp. 229–261, 2005.

[9] P. C. Kanellakis and S. A. Smolka, "Ccs expressions finite state processes, and three problems of equivalence," *Inf. Comput.*, vol. 86, no. 1, pp. 43–68, May 1990.

[10] S. M. Kristek and D. A. Shell, "Orienting deformable polygonal parts without sensors," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2012.

[11] G. Laguna, R. Murrieta-Cid, H. M. Becerra, R. Lopez-Padilla, and S. M. LaValle, "Exploration of an unknown environment with a differential drive disc robot," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2014.

[12] S. M. LaValle, *Planning algorithms*. Cambridge University Press, 2006.

[13] S. M. LaValle *et al.*, "Sensing and filtering: A fresh perspective based on preimages and information spaces," *Foundations and Trends® in Robotics*, vol. 1, no. 4, pp. 253–372, 2012.

[14] R. Lopez-Padilla, R. Murrieta-Cid, and S. M. LaValle, "Optimal gap navigation for a disc robot," in *Algorithmic Foundations of Robotics X*. Springer, 2013, pp. 123–138.

[15] R. Milner, "A calculus of communicating systems," *Lecture Notes in Comput. Sci. 92*, 1980.

[16] J. M. O'Kane and D. A. Shell, "Automatic reduction of combinatorial filters," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2013.

[17] ——, "Concise planning and filtering: Hardness and algorithms," *IEEE Transactions on Automation Science and Engineering*, 2017.

[18] R. Paige and R. E. Tarjan, "Three partition refinement algorithms," *SIAM Journal on Computing*, vol. 16, no. 6, pp. 973–989, 1987.

[19] D. Park, "Concurrency and automata on infinite sequences," in *Theoretical Computer Science*. Springer, 1981, pp. 167–183.

[20] J. Rot, M. Bonsangue, and J. Rutten, "Proving language inclusion and equivalence by coinduction," *Information and Computation*, vol. 246, pp. 62–76, 2016.

[21] J. J. Rutten, "Automata and coinduction (an exercise in coalgebra)," in *Int. Conf. on Concurrency Theory*. Springer, 1998, pp. 194–218.

[22] ——, "Universal coalgebra: a theory of systems," *Theoretical computer science*, vol. 249, no. 1, pp. 3–80, 2000.

[23] F. Z. Saberifar, S. Ghasemlou, J. M. OKane, and D. A. Shell, "Set-labelled filters and sensor transformations," in *Proc. Robotics: Science and Systems*, 2016.

[24] F. Z. Saberifar, A. Mohades, M. Razzazi, and J. M. O'Kane, "Combinatorial filter reduction: Special cases, approximation, and fixed-parameter tractability," *Journal of Computer and System Sciences*, vol. 85, pp. 74–92, 2017.

[25] D. Sangiorgi, "On the origins of bisimulation and coinduction," *ACM Transactions on Programming Languages and Systems (TOPLAS)*, vol. 31, no. 4, p. 15, 2009.

[26] B. Tovar, *Minimalist models and methods for visibility-based tasks*. University of Illinois at Urbana-Champaign, 2009.

[27] B. Tovar, F. Cohen, L. Bobadilla, J. Czarnowski, and S. M. Lavalle, "Combinatorial filters: Sensor beams, obstacles, and possible paths," *ACM Transactions on Sensor Networks (TOSN)*, vol. 10, no. 3, p. 47, 2014.

[28] B. Tovar, R. Murrieta-Cid, and S. M. LaValle, "Distance-optimal navigation in an unknown environment without sensing distances," *IEEE Transactions on Robotics*, vol. 23, no. 3, pp. 506–518, 2007.

[29] J. Van Benthem, "Modal correspondence theory [ph. d. thesis]," *University of Amsterdam, Netherlands*, 1976.

[30] A. Van der Schaft, "Equivalence of dynamical systems by bisimulation," *IEEE transactions on automatic control*, vol. 49, no. 12, pp. 2160–2172, 2004.

[31] J. Yu and S. M. LaValle, "Story validation and approximate path inference with a sparse network of heterogeneous sensors," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2011.

[32] ——, "Shadow information spaces: Combinatorial filters for tracking targets," *IEEE Transactions on Robotics*, vol. 28, no. 2, pp. 440–456, 2012.