

# A Comparison of Multi-agent State-space Exploration Approaches for the Verification of Mixed and Partially Observable Domains

Galina Novikova<sup>[0000-0002-4949-9135]</sup> and Esteban Azoifeifa<sup>[0000-0003-0604-4234]</sup>

RUDN University, Moscow, Russia

**Abstract.** Validation problems concern practically any domain theory that describes the types and properties of objects, classes of states and relations, and the laws of functioning and interaction between objects. Recognition and verification of the state space associated to a determined system is key to solving management problems. Such tasks in the context of state-based systems often require a reachability analysis that displays an exponential expansion of the search space in certain domain theories. Coordinated and decentralized multi-agent exploration can offer efficient solutions to the problem, but the task becomes intractable within certain probabilistic domains. Concretely, the associated 'curse of dimensionality' of partially observable domains, in practice, is a considerable limitation that allows to tackle the recognition task only in terms of suboptimal solutions. However, the signals associated to properties in an environment can be read by different kinds of sensors, which in turn can significantly influence the state exploration time in the recognition task. In this paper, we study multi-agent systems for state-space exploration in environments with partial and mixed observability, and we outline certain conditions under which interchangeability between approaches can be obtained according to efficiency goals. We performed a cost comparison of two multi-agent approaches consisting of automated agents without inner states, with the difference that one of them uses exact, noise immune sensors and the other utilizes unbiased sensors with a certain degree of imprecision. By representing the interpretation of a domain as a path exploration task, we obtained an equation comprising the necessary conditions for an equivalence between the approaches.

**Keywords:** Verification, uncertainty, domain, multi-agent, partially observable, reachability, search space.

## 1 Introduction

Creating tools for recognition and verification of object states is a key task in the development of intelligent control systems for complex dynamic systems. The solution for this problem lies in the field of research and development of multi-agent systems, but the problem becomes unsolvable in certain probabilistic areas. In particular, the so-called 'curse of dimensionality' of partially observed domains can become a significant limitation for the search of optimal solutions to the task. We consider that the

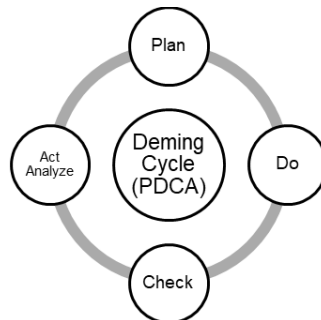
Proceedings of the XXII International Conference "Enterprise Engineering and Knowledge Management" April 25-26, 2019, Moscow, Russia

solution to the recognition problem should be based on a formal approach, in relation to both the definition of the concept of 'state' and the formalization of the recognition process.

The concept of 'state' is fundamental in the field of control theory, since control, in a broad sense, is the definition and application of actions in order to change or preserve the state of an object. Thus, one of the most important tasks arising in the control process is the recognition and verification of the states associated to the objects in a domain or subject area for the correct application of control actions.

State changes occur according to a specified control objective. The purpose of the control system can be regulation: to guarantee that the object will function within a given state or, more precisely, within a given class of states. If we consider a dynamic system—such as an enterprise— as our subject area, the problem of regulation is solved at the level of operational management. On the other hand, the task of developing an action that will change the state of the control object in order to achieve some target state is solved at the level of strategic management [1].

In the field of corporate management, the technology known as management by exception is widely used: the Deming management cycle [2], for example, includes a stage to control and verify deviations of real values from the planned indicators (Fig. 1). This approach can be used for the purpose of management in any complex dynamic system, which in turn can be attributed to the class of quasi-stationary objects [3].



**Fig. 1.** Tasks in the Deming Cycle

We understand as a quasi-stationary object a dynamic system whose behavior is determined by the state transitions of its elements and the system as a whole followed by stationary periods  $\Delta t$ , during which the state of the system is constant. It can be assumed that the minimal  $\Delta t$  is determined by the time of the minimal transaction, which nowadays is increasingly being reduced with the development of novel technologies in telecommunications and other engineering-related fields.

## 2 State Space

A description of the behavior of the systems should include the possible state transition sequences of the individual elements and the system as a whole; concretely, its cur-

rent status and the planned, possible, probable or objective states, among others. The state space includes various classes of states that can be selected:

- The classes of valid states ( $\Omega_v$ ).
- The classes of conflicting states that require the development of control actions ( $\Omega_c$ ).
- The classes of objective states ( $\Omega_o$ ).

Object states can be determined by observing the compliance of the instances to some boundaries on the property values of the object, e.g. 35 to 41 degrees for the property 'temperature'. Additionally, states can be distinguished by the presence of a determined property value of an object –especially if it is not quantifiable (e.g. the color red)– or by a process of change (adaptation) of this parameter.

In this work, we assume that the state of the system as a whole is determined by the states of its elements. Concretely, control objects in a complex dynamic system correspond not only to single objects of the subject area, but also to the connections and relationships between objects. An example is the case of enterprise architecture development, where links between objects can give rise to a new object: a set of related positions in an organization forms a department, an organizational unit of a higher order. In such cases, identification of a global system state requires the recognition of both states, connections and relationships between objects.

A simple connection of some kind between objects can be 'strong', 'weak', 'planned', 'undesirable'. On the other hand, states associated to relations such as 'part-whole' and 'element-set' could also be used to represent an inheritance mechanism of object states, case in which we would be talking about relationships. In particular, we consider that relationships between objects form objects of higher order, which can extend its properties and states to its constituent elements (inheritance) but can also display new properties and states based on the principle of 'emergence' [4]. For example, a 'sales' relation, which includes objects such as 'product', 'customer', 'sales channel', can present states like 'low', 'high', and 'unprofitable'. The states are identified by the values of the properties of the object: profitability, profit, volume.

The approach in [5] proposes the use of predicate calculus to describe domain theories in a simplified form. In detail, a formal system  $M$  can be expressed in the following way:

$$M = \langle D, F, A, I \rangle,$$

where  $D$  is the theory alphabet (a finite set of basic characters);  $F$  is a set of (countable) formulas (also called well formed formulas of *wff*), built from elements of  $D$  using some set of syntax rules;  $A$  corresponds to a set of formulas called axioms, and  $I$  is a finite set of inference rules. Axioms constructed using the alphabet of the theory establish the correspondence between:

- Object type ( $T$ ) and the set of properties ( $P$ ).
- Types of objects and the set of relations ( $R$ ).
- Object type and the set of possible object states ( $S$ ).

Additionally, the statements about properties [6] and relations between objects described by axioms are accepted as true without the need of a proof. Finally, the threshold or current values of object properties are defined with the help of a set of constants.

The current state of a domain is identified by the set of atomic axioms of the formal system characterized by terms with a constant value, as well as bound variables fixed by the equality predicate, which in turn correspond to terms of a subset of the atomic axioms. Thus, the complete set of states is obtained from atomic axioms with constants as terms, as well as axioms describing cause-effect, structural and functional relations between the elements of the domain. In particular, axioms that reflect the structural relations of the domain elements, as well as axioms that describe inheritance, can be used along with the mechanism of logical inference to expand the set of facts that identify the current state.

### 3 Formalization of a State

For the formal definition of the classes of states we use the model-theoretic approach, dividing the whole state space into subsets of equivalent state classes, each of which is described by a closed (ground) logical formula. Such formula can include atomic statements with constant terms, as well as statements in which free variables are bound by a quantifier: in general, the expressive power of the theory alphabet determines the degree of uniqueness of the statements about domain properties and state classes. For example, not only traditional quantifiers ( $\forall, \exists$ ) can be used to describe a class of states, but also other ones such as 'there exists one and only one' ( $\exists!$ ) and 'there is no' ( $\nexists$ ), as well as modal operators such as 'possible', 'necessary', 'allowed' and 'forbidden' in addition to operators reflecting temporal and spatial modalities [7], among others.

Let us now present a relational system  $G = \{Q, W\}$ , where  $Q$  is the carrier or underlying set of elements of the system, and  $W$  is the set of relations  $w_j \subseteq Q^{m_j}$  ( $j \in J$ ) between the elements in  $Q$ , where a predicate  $Z_j(q^*) = True$  indicates that the element  $q^* = (q_1, q_2, \dots, q_{m_j}) \in w_j$  with arity  $m_j$  defined on the carrier set by a function  $ar: Y_z \rightarrow \mathbb{Z}^+$ ,  $Y_z = \{Z_j \mid j \in J\}$ . Thus, if a complex dynamic system functioning under a formal system  $M$  happens to be in one of the states ( $s_g \in S_g$ ) of  $G$ , the behavior of the system can now be described by a set of relational systems, and in this case the values that bound the free variables will determine the properties of the system.

We first distinguish the classes of equivalent relational systems [8] and the corresponding classes of equivalent states containing the elements of the domain, and then we associate to each class a closed logical formula defined in the signature of the system and taking the value 'true' or 'false'. Concretely, the formal system describing the current state of the domain corresponds to a model of one of the equivalence classes,  $\Gamma$ , which in this case contains all relational systems of signature  $\sigma = (Y_z, ar)$ . The formal system, in turn, yields an interpretation under which the closed formula identifying the class takes the value 'true' [9], and the formula can then be considered as an assertion about the properties of the model. Accordingly, the entire set of formal sys-

tems describing the behavior of a complex dynamic system is divided into classes of models, each of them associated to a closed logical formula describing the corresponding class of states ( $\Omega_v$ ,  $\Omega_c$  or  $\Omega_o$ ).

This approach can be applied in the engineering phase or in the everyday control and monitoring operations in a complex organization. The engineering of a company's processes, for example, should include guarantees that the system will avoid or minimize its state transitions to conflicting states ( $\Omega_c$ ) while guaranteeing that it will reach objective states ( $\Omega_o$ ) in a cost-effective way. Control and monitoring, or anti-crisis operations, are destined to take the system out of those undesirable states. On the other hand, strategic management deals with finding new ways to maximize the organization's revenue in a competitive environment with other players that share the same goals. This is expected to be accomplished by finding the correct combination of actions that maximize the frequency of encountering organizational states inside the objective set ( $\Omega_o$ ).

If these processes are considered as the same optimization task in different manifestations, then the benefit of this approach can be notably perceived in the automation of such processes. Concretely, a software tool can predict the transitions into conflictive or undesirable states (or objective states) by looking at all the possible ways that their logical formulas can be bound, taking into account the formalized relational systems, historical data and/or experts' knowledge. Such predictions, which fall outside the scope of this work, can be obtained from a probabilistic model of state transitions based on historical data, for example, such that the corresponding anti-crisis (or strategic) actions can be taken even before an undesirable state is reached or within an optimal time threshold after its parameters have been identified. This automation yields a reduction in financial, material, technical and informational costs.

## 4 Approaches to State-space Exploration

Verification of a domain to determine the corresponding classes of states is preceded by the task of recognizing all the possible object instances in order to bound the free variables. Having outlined the verification task, in this work we focus on the recognition task, and in particular on the analysis of efficient methods of environment observation taking into account the 'curse of dimensionality' in state-space exploration under resource constraints.

Concretely, we explore the conditions under which it is feasible or even recommended that very expensive sensors with a perfect capacity of observation (noise immune [10] and perfect accuracy) be replaced in multi-agent recognition tasks with more affordable sensors with a limited capacity of detecting precise signals but with the ability to guarantee a minimal observation confidence.

We begin by defining a set  $P$  of possible (perceivable) properties of objects in a domain, so that each possible combination of properties is uniquely associated to an object type in  $T$ . In the initial case of full observability, the agents conforming the recognition multi-agent system are equipped with  $|P|$  sensors, each of them capable of detecting the value of one of the properties with full accuracy and precision.

By means of the threshold function  $th: P \rightarrow V$ , we associate every property with a numerical range from the set  $V$  that contains all the possible values that the property can take. The set of values is built in the following way:  $V = \{iv(a,b) \mid \forall a, b \in N, a \leq b\}$ , where  $N$  denotes the set of natural numbers. Every range is constructed by the function  $iv(a,b) = \{x \in N \mid a \leq x \leq b\}$ ,  $a, b \in N, a \leq b$ .

The process of binding free variables to their corresponding values or quantifiers is achieved by the functions  $l_T$  and  $l_S$ , which correspond to the labelings of object types and object states respectively. Both  $l_T(X): Pow(X) \rightarrow 2^\psi$  and  $l_S(X): th(p^1) \times th(p^2) \times \dots \times th(x^{|X|}) \rightarrow 2^\psi$  map to the set of propositions  $\psi$  in  $M$ , with  $Pow(X) = \{X^*/X^* \subset X\}$  referring to the powerset of the set  $X$ .

Once physically or virtually within a location  $g$  of an environment grid, which contains an undetermined number of objects and connections and relationships between them, an agent can perform an observation represented by the vector  $obs(X, g) = (x_1, x_2, \dots, x_i, \dots, x_{|X|})$ ,  $x_i \in X_i, X_i \in X$ . For simplicity, we assume that the closest approximation to the real parameters of a domain presented as an infinite grid is obtained by observing the maximum possible number of locations independently of the exploration path, which implies homogeneity of the grid in terms of the hidden characteristics of the domain and the possibility to use parallel multi-agent exploration to keep the recognition time less than  $\Delta t$ .

Once an observation has been made, a set of tuples summarizing the collected information can be obtained by the function  $dv(x^*): \{\langle X, Y, l_T(X) \rangle \mid (X \in Pow(\{x^*\}))\}$ , where  $X$  corresponds to the set of object properties,  $Y$  the associated observed values and the third term is the labeling of  $X$ . Before any concrete implementation in a subject area, our approach aims to enumerate all the possible combinations of object types and values by means of the powerset function. This leads to a function that keeps within the analysis the worst case scenario in terms of complexity.

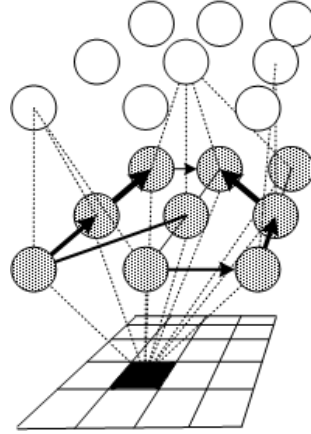
$$O(a^*) = \bigcup_{\langle X, Y, l^* \rangle \in dv(a^*)} l_s(Y) \quad (1)$$

Equation 1 yields all the possible object instances present in a location  $g$  according to the sensor inputs. This set, however, corresponds only to the first input in the process of determining the totality of objects and relations in the location, considering that connections between object instances and relationships of higher order are to be discovered after the initial signal perception.

Theoretically, an infinite number of connections can be established between two entities, making it impossible to conclude a verification task. In our model, we reduce the recognition time of a binary relation to a constant ( $t = 1$ ), and make it possible to delegate the deterministic interpretation (labeling) task to either one or multiple intelligent agents according to the desired multi-agent exploration strategy. However, the interpreting agents need to explore not only the totality of the recognized elements in the domain, but also the totality of the possible recognition sequences (paths) between entities in order to reveal the complete set of connections and relationships.

We project such interpretation task as a recursive process among different orders or levels, beginning with objects and relations and continuing with second and higher

order relationships. Nonetheless, we assume that all objects and relations on a certain interpretation level must be recognized before entering a higher order level. Additionally, we presuppose that all entities on a certain interpretation level can be discovered without the need to extend the sequence into a higher order.



**Fig. 2.** Multi-level interpretation process on a grid location of a domain

Under these conditions, the interpretation process turns into the enumeration of all possible paths from the elements yielded in the previous level to every potential destination on that level (Fig. 2). As such, the first iteration takes as inputs the set of detected entities from the perceived signals, while the next iterations take as input the set of explored paths on the last level.

$$cc(n) = \sum_{k=1}^n {}^n P_k = \begin{cases} n(cc(n-1) + 1) & ; n > 0 \\ 0 & ; otherwise \end{cases} \quad (2)$$

Representing input elements as vertices in a complete undirected graph [11], Equation 2 enumerates all the possible paths between the elements and all the potential destinations. The equation makes use of the recursive function  $cc_{(k)}(n) = cc(cc_{(k-1)}(n))$  with  $cc_{(0)}(n) = n$ , where the result of the actual level is fed as an input to the interpretation process (path enumeration) on the next level. This results in the total path quantity in  $h$  levels by means of the function  $Path(A, h) = \sum_{k=1}^h cc_{(x)}(|A|)$ .

## 5 Partial and Mixed Observability

Until this point we have studied the case when agents perform an observation  $obs(g)$  with perfect sensor accuracy, meaning that the agents are required to perform no more than one  $obs(g)$  in order to get the true value of the signals underlying the real state of the system. The case of partial observability [12], on the other hand, ends with this

assumption and introduces uncertainty (noise) on the signal measurements, which in turn can compromise the domain verification process.

We compare the full observability case with agents whose sensor measurements are unbiased –maintain a maximal accuracy– but display measurement errors that are normally distributed. The assumption of normality on the noise of the signals with full accuracy in respect to the true average of the data allows the agent to repeatedly perform a measurement in order to improve the sensor precision, represented by the inverted variance  $\tau = 1/\sigma^2$ .

Another approach that does not involve repeated interaction with the environment corresponds to the extension of the single observation  $obs(g)$  into a range of hypotheses of the possible real values according to the non-modified precision of the sensors. The agent then tests the hypotheses and determines whether they result in the same interpretations or in completely different ones, case in which the first approach (repeated measurement) should be preferred.

$$eps(g) = (X_1, X_2, \dots, X_{|P|}), X_i = \{x \in th(p_i), p_i \in P \mid x_i - \zeta \leq x \leq x_i + \zeta\} \quad (3)$$

where  $x_i \in obs(g)$  and  $\zeta \in N$ . Equation 3 takes the initial observation as the central value and modifies the tuple with the addition of a range of size  $\zeta$  around  $obs(g)$ . Such approach can be feasible for low interpretation orders, as the interpretation state-space grows exponentially bigger with the addition of new levels and can quickly become intractable even for a large multi-agent system working in parallel.

Thus, the newly generated ranges yield the new observation space  $ss(X^*) = \{(x_1, x_2, \dots, x_i, \dots, x_{|X^*|}) \mid \forall x_i \in X_i, x_i \in X^*, \text{ waiting to be tested by the agent. However, because of the 'curse of dimensionality' it could be impossible to test every hypothesis, and the only feasible alternative would be choosing a subset of the observation space to perform sampling over the } sp(X^*) = |X_1| * |X_2| * \dots * |X_{|X^*|}| \text{ different states to reach conclusions with a lower level of confidence.}$

If there is no difference in the cost of assuming multi-agent verification with noise immune sensors versus the error prone approach, then the immediate option to guarantee an error free recognition of a domain environment would be the first one. However, such superior sensing capabilities could mean in real life an elevated and even prohibitive cost which would force a change of method.

$$\beta_a \cdot Path(O(obs(P, g)), h) + \mu = \beta_b \cdot \sum_{ob \in ss(eps(g))} Path(O(ob), h) \quad (4)$$

Equation 4 refers to the necessary conditions for an equivalence in resource expenses between the full observability exploration (left) and the partial observability one. Constants  $\beta_a$  and  $\beta_b$  denote the unitary cost of performing an interpretation activity like searching for the next node in a path. On the other hand,  $\mu$  represents the additional cost that the optimal sensors represent with respect to the lower-quality ones, and it can be factored as  $\mu = (\beta_b - \beta_a) * \gamma$ , where  $\gamma = \sum_{ob \in ss(eps(g))} Path(O(ob), h) = \eta \cdot Path(O(obs(P, g)), h)$ .



Concretely, the left part of Equation 4 (full observability) yields a single path quantity whereas the right part (partial observability) yields a field of possible paths (an uncertainty range) in the form of a summation of the function *Path*. This path space can be exponentially bigger than the full observability space set, depending on the size of the set  $P$  of properties and its associated set of values, as well as the number  $h$  of higher order levels to take into account. The balancing factor  $\mu$  is specific to each set of optimal sensors, and consequently to the set of properties being analyzed. In particular, this function serves the purpose of yielding an optimal mixed observability set of sensors; however, this is the case only when the quality of at least one of the optimal sensors is greater than the quality obtained when performing multiple observations with normal sensors at the same cost of the optimal sensor. A middle point between both approaches corresponds to a significant improvement over the computationally heavy method and the costly optimal one: this is the case of mixed observability [13], the replacement of only a subset of the sensors for high-precision ones, which promises a balance between resource expense and efficiency. Even though the partial observability of the domain states will not be eliminated after the implementation of such approach, the search space for path exploration is guaranteed to get substantially trimmed. Concretely, the search space gets reduced by a factor of  $1/|P|$  for each sensor introduced.

## 6 Conclusion

Recognition and verification of the state space associated to a determined system is key to solving management problems. Of particular interest is the task of recognizing the current state of object instances in a domain and assigning it to one of the possible state classes, which in turn can be associated to a certain control or regulation solution.

However, the variety of objects, properties, connections and relationships in complex structured dynamic domains provoke the problem known as 'curse of dimensionality', which directly affects finding optimal methods for the verification task. We consider necessary to integrate different types of agents and to apply parallelization in order for the recognition time to be less than  $\Delta t$ , the lapse in which the system and its elements remain in the same state before making a transition to another state.

In order to achieve this goal, however, different approaches to performing recognition tasks can be implemented. In particular, the signals associated to properties in an environment can be read by means of different kinds of sensors, which in turn can significantly influence the state exploration time in the recognition task. Assuming the utilization of one kind of sensor per object property and one sensor per agent, we compare the possibility of using exact, noise immune sensors versus the alternative of utilizing unbiased sensors with a certain degree of imprecision that yields normally distributed errors. By representing the interpretation of a domain as a path exploration task, we obtained Equation 4 with a coefficient  $\gamma$  as the basis to compare the cost-effectiveness of both sensing methods, and finally arrived at a mixed approach capable of significantly trimming the search space while remaining cost-effective.

This approach is meant to be applied in the engineering phase or in control and monitoring operations in a complex organization in order to reduce financial, material, technical and informational costs. This recognition method is assumed to be performed by automated agents without inner states, or whose inner states, defined by the agent's own properties, do not affect the states of the object or relation to be recognized: as such, agents' internal structure is left out of the model due to its irrelevance in relation to the recognition task. Additionally, computational solutions can be developed in order to predict the system's transitions into conflictive or undesirable states (or objective states) by looking at all the possible ways that their logical formulas can be bound. These automation methods can be complemented with the addition of probabilistic forecasts based on historical data.

## References

1. Novikova, G.M., Azofeifa, E.J.: Semantics of big data in corporate management systems. *RUDN Journal of Mathematics, Information Sciences and Physics* 26(4), 382-391 (2018).
2. Singh, J., Singh, H.: Continuous improvement approach: state-of-art review and future implications. *International Journal of Lean Six Sigma* 3(2), 88-111 (2012).
3. Repin, V.V., Eliferov, V.G.: Process approach to management. *Business process modeling. Standards and quality* (2008).
4. Krasnogor, N., Smith, J.: Emergence of profitable search strategies based on a simple inheritance mechanism. In: *Proceedings of the 3rd Annual Conference on Genetic and Evolutionary Computation*. pp. 432-439. GECCO'01, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2001).
5. Novikova, G.M., Azofeifa, E.J.: Domain Theory Verification Using Multi-agent Systems, pp. 120-125. *Procedia Computer Science* (2016).
6. Kaneiwa, K., Mizoguchi, R., Nguyen, P.H.P.: A logical and ontological framework for compositional concepts of objects and properties. *New Generation Computing* 33(2), 149-172 (2015).
7. Blackburn, P., Rijke, M.d., Venema, Y.: *Modal logic*. Cambridge University Press (2001).
8. Chajda, I., Langer, H.: Quotients and Homomorphisms of Relational Systems, *Acta Univ. Palacki. Olomuc., Fac. rer. nat., Mathematica* 49, 37-47 (2010).
9. Novikova, G.M.: A conceptual approach to the design of intelligent control systems. In: *Proceedings of the III International Symposium "Intellectual Systems" INTELS 98*, pp. 49-51 (1998).
10. Fomina, M., Eremeev, A., Vagin, V.: Noise models in Inductive Concept Formation, pp. 413-419. *Lecture Notes in Business Information Processing* (2013).
11. Addario-Berry, L., Broutin, N., Goldschmidt, C., Miermont, G.: The scaling limit of the minimum spanning tree of the complete graph. *The Annals of Probability*, 45(5), 3075-3144 (2017).
12. Cai, K., Zhang, R., Wonham, W.M.: Relative observability and coobservability of timed discrete-event systems. *IEEE Transactions on Automatic Control* 61(11), 3382-3395 (2016).
13. Ong, S.C.W., Wei Png, S., Hsu, D., Lee, W.S.: Planning under uncertainty for robotic tasks with mixed observability. *The International Journal of Robotics Research* 29(8), 1053-1068 (2010).