A-EMS: An Adaptive Emergency Management System for Autonomous Agents in Unforeseen Situations

Glenn Maguire, Nicholas Ketz, Praveen Pilly, Jean-Baptiste Mouret

A-EMS: An Adaptive Emergency Management System for Autonomous Agents in Unforeseen Situations

Glenn Maguire[1], Nicholas Ketz[2], Praveen K. Pilly[3], and Jean-Baptiste Mouret[1]

[1] Inria, CNRS, Université de Lorraine; {glenn.maguire, jean-baptiste.mouret}@inria.fr
[2] Colossal Biosciences; nick.ketz@gmail.com
[3] Proficient Autonomy Center, Intelligent Systems Laboratory, HRL Laboratories; pkpilly@hrl.com

Abstract. Reinforcement learning agents are unable to respond effectively when faced with novel, out-of-distribution events until they have undergone a significant period of additional training. For lifelong learning agents, which cannot be simply taken offline during this period, suboptimal actions may be taken that can result in unacceptable outcomes. This paper presents the Autonomous Emergency Management System (A-EMS) - an online, data-driven, emergency-response method that aims to provide autonomous agents the ability to react to unexpected situations that are very different from those it has been trained or designed to address. The proposed approach devises a customized response to the unforeseen situation sequentially, by selecting actions that minimize the rate of increase of the reconstruction error from a variational autoencoder. This optimization is achieved online in a data-efficient manner (on the order of 30 to 80 data-points) using a modified Bayesian optimization procedure. The potential of A-EMS is demonstrated through emergency situations devised in a simulated 3D car-driving application.

Keywords: Adaptive Control · Intelligent Robotics · Lifelong Learning.

1 Introduction

There has been much progress in recent years in machine learning algorithms that enable autonomous agents to learn how to perform tasks in complex environments online based on observations and sensor feedback. Recent advances in Reinforcement Learning (RL) through deep neural networks in particular have shown promising results in developing autonomous agents that learn to effectively interact with their environments in a number of different application domains [3, 11], including learning to play games [17, 6], generating optimal control policies for robots [20, 21], speech recognition and natural language processing [4], as well as making optimal trading decisions given dynamic market conditions [8]. Under the RL paradigm, the agent learns to perform a given task through numerous training episodes involving trial-and-error interactions with its environment. By discovering the consequences of its actions in terms of the rewards obtained through these interactions the agent eventually learns the optimal policy for the given task.

These approaches work well in situations where it can be assumed that all the events encountered during deployment arise from the same distribution on which the agent has been trained. However, agents that must function within complex, real-world environments for an extended period of time can be subjected to unexpected circumstances outside of the distribution they have been designed for or trained on, due to environmental changes that arise. For example: an autonomous driving car may encounter significantly distorted lane-markings that it has never experienced before due to construction or wear, and must determine how to continue to drive safely; or an unaware worker in a manufacturing facility may suddenly place a foreign object, such as their hand, within the workspace of a vision-guided robot-arm that must then react to avoid damage/injury. In such unexpected, novel situations the agent's policy would be inapplicable, causing the agent to perhaps take unsafe actions.

In this paper, we consider scenarios where a trained agent encounters an unforeseen situation during deployment that renders available system or state-transition models highly unreliable, so that any inferences based on such models, as well as any pre-defined safe state/action regions, are no longer valid for safe decision-making. An agent unable to respond effectively to a novel situation when first encountered is vulnerable to take dangerous actions. This is of particular concern in safety-critical applications where sub-optimal actions can lead to damage or destruction of the agent or loss of human life.

We address this problem by developing a data-driven response-generation system that allows an agent to deal with novel situations without reliance on the accuracy of existing models, or the validity of safe states and recovery policies developed offline or from past experiences. The key insight to our approach is that uncertainty in observations from the environment can be used as a driver for the generation of effective, short-term responses online, when necessary, to circumvent dangers, so that the agent can continue to function and learn within its environment.

Increased observation uncertainty has been used in the past to detect novelty (e.g., by measuring out-of-bounds auto-encoder reconstruction errors [23]), indicating situations for which the existing policy is unprepared. It stands to reason, then, that decreasing this uncertainty would decrease novelty and return states to those that the current policy can handle effectively. The work in this paper investigates, therefore, how uncertainty-minimization can be correlated with safe and effective actions in situations where the existing policy would fail. While use of uncertainty to detect potential danger is not new, using it to generate actions in an online manner, customized to the particular never-before-seen emergency as it unfolds, is novel.

In the absence of a reliable model or policy network to make proper action decisions, determination of an appropriate response to a novel situation must be data-driven and sequential. Moreover, in an emergency situation this response must be devised efficiently (i.e., in just a few time-steps), meaning that little data will typically be available for finding the optimal actions to take. This reactive approach, therefore, necessitates a fast, online, optimal decision-making method.

Bayesian Optimization (BO) provides an ideal theoretical framework for this type of problem [18]. BO is a data-efficient, global-optimization method for sequential decision-making where the objective function is expensive to evaluate or is taken to be a black-box. It builds and sequentially improves a probabilistic model of this objective through measurement data obtained online. This model is used to compute the next best action to take in a manner that balances exploration of the unknown regions of the objective and exploitation of regions found to be most likely to contain the optimal value.

Using this framework we devise an emergency-response-generation method that combines a modified BO procedure for efficient sequential optimization, with Gaussian Process (GP) regression for representing the probabilistic model of the objective. The objective function in our approach is a metric designed to capture the uncertainty in the observations obtained by the autonomous agent in a way that facilitates the generation of an effective emergency response. The responses generated by this method are intended to be action-sequences over a short time-span that are only initiated when deemed necessary to circumvent a dangerous situation that the agent is not yet prepared to handle. Our approach is referred to as the Autonomous Emergency Management System (A-EMS).

2 Related Work

Existing works related to safety for autonomous agents typically involve incorporating pre-designed penalties into the reward or cost function for actions deemed unsafe when training a deep neural network to generate policies [25, 2], or restricting agent actions to "safe" regions to prevent it from reaching unsafe states [10, 27]. Other approaches use examples of dangers in offline training in representative environments to either help identify conservative behaviors to use based on pre-specified rules [23], or to learn recovery policies for specific dangerous scenarios [26]. However, significantly novel events can arise in complex environments that produce dangerous scenarios not accounted for through the above-mentioned mechanisms, thereby requiring a customized response.

An agent must therefore be able to continually learn and adapt to such novel situations. Continual learning approaches in the literature, though, do so through the initiation of a new learning phase [7, 19]. Adaptation to the novel situation, then, is not instantaneous, and must happen over an extended period of time dictated by the continual learning method used.

Nevertheless, what existing approaches do show is that deep learning neural networks produce erratic and unreliable predictions when presented with inputs very different from their training scenarios [23, 24], but also that uncertainty in predictions from such out-of-distribution inputs can be an effective way to detect novelty [10, 23, 15]. Moreover, trying to jointly optimize for task performance and safety-violation can lead to restrictive, sub-optimal policies [26, 1].

In addition, despite their limitations, these prior works also make it clear (see, for example, [27]) that including a safety mechanism to assist learning agents improves success-rate, constraint satisfaction, and sample efficiency.

3 Problem Description

We consider the A-EMS method to be used as part of an independent module that monitors a trained and deployed agent as it performs a given task. Within this scenario there may be instances where the agent encounters a situation it has never seen before that presents a danger if not acted upon properly. The agent's existing policy is unable to determine an appropriate response without further training, and any environment models become unreliable. Whenever such an unforeseen event is encountered, the agent is considered to be in an emergency situation for which an emergency response is required to mitigate the danger.

It is assumed that an emergency detection method is available that monitors the agent during deployment and can identify a novel situation that the agent is unprepared to handle. Numerous approaches to novelty detection can be found in the literature (e.g., [23, 7, 16]) which can be used for this purpose. Moreover, context-specific information relevant to the given application domain could also be used to further verify that the agent is in imminent danger if it continues with the actions output by its existing policy.

While monitoring, the A-EMS method remains disengaged and the agent is allowed to freely perform its task using its existing policy. Once an emergency situation is found to be imminent, the A-EMS response generation algorithm is engaged (Figure 1), which takes over the policy's actions by replacing them with a suitable emergency response. Consequently, this necessitates the halting of any updates to the existing policy over the course of the response.

The response devised is a customized action-sequence over the next N time-steps to address the danger. This action-sequence must be generated online as the encounter with the novel situation unfolds.

4 Methodology

The proposed method for generating a response to address an unforeseen situation is based on the idea that taking actions that reduce uncertainty in the observations should correspond to an effective response that guides the agent to a more familiar state, which the existing policy knows how to effectively handle. In our approach, the uncertainty associated with novel situations is represented by the reconstruction errors (i.e., mean squared pixel-value errors) from a Variational Auto-Encoder (VAE) [13] designed to process RGB images from a camera sensor. Its neural network structure was borrowed from that presented in [12].

The response devised by A-EMS for an emergency situation is an action-sequence that spans some fixed number of time-steps, N. The generation of actions that reduce observation uncertainty is thus taken to be a sequential

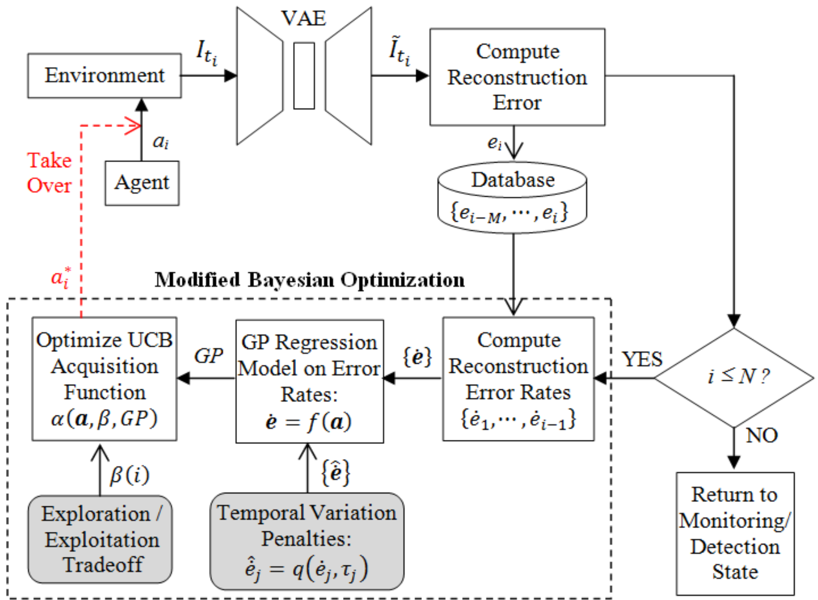

Fig. 1: Workflow of the Autonomous Emergency Management System.

optimization process, where each action must ideally be the optimal decision to make given all the data gathered since the initiation of the response.

To perform this online optimization we use BO coupled with GP regression. BO is a data-efficient technique to find the global optimum of a function, $f(\mathbf{x})$, that is significantly expensive to evaluate. It achieves this by building a surrogate, probabilistic model, $G[f(\mathbf{x})]$, of f, which includes a mean function, $\mu(\mathbf{x})$, representing the current best estimate of $f(\mathbf{x})$ over the domain of f, and a variance function, $\sigma^2(\mathbf{x})$, representing the uncertainty in this estimate. We herein employ GP regression to construct this surrogate model [22].

Using this GP model, BO optimizes a corresponding, and relatively simpler, heuristic function, $\alpha(x, G[f(\mathbf{x})])$, termed the acquisition function, which quantifies the utility of any given input, x, in terms of its potential for optimizing $f(\mathbf{x})$. This optimization is achieved by sequentially sampling inputs from the domain of x that have the greatest potential for optimizing f as indicated by the acquisition function. More details on BO can be found in [18].

As shown in Figure 1, the sequential optimization uses the rate of change of the VAE reconstruction errors to drive the BO loop at each time-step, i, of the response action-sequence. This is because there may be situations where it may not be possible to find actions that reduce the reconstruction errors, and all that can be done is to minimize its increase. This would still be a valid response if that is the best that can be done given the circumstances. An imminent collision with an obstacle is a good example – some situations may simply call for maximum braking as there may not be any way to swerve around the obstacle. In such cases, errors would only rise as the agent approached the obstacle, with braking helping to slow down the rate of increase until it eventually plateaus at a higher but stable value. Minimizing the error-rate would capture the need to slow down the rise of the errors in such situations, but would also be able to keep driving the errors down further (i.e., negative error-rates) if it is indeed possible.

The objective at each time-step, $i \in [1, N]$, of the response is to find an action, a_i^*, that minimizes the error-rate, $\dot{e}_i$, that would result from that action, by conducting one cycle of the BO loop shown in Figure 1. This optimization will have available data-points containing all the actions, $\mathbf{Pa}_i = \left[a_1\ a_2\ ...\ a_{i-1}\right]_i$, taken in the last $(i-1)$ time-steps of the action-sequence, as well as the corresponding true error-rates, $\mathbf{P\dot{e}}_i = \left[\dot{e}_1\ \dot{e}_2\ ...\ \dot{e}_{i-1}\right]_i$, that resulted.

The last M error data-points are always stored in a database. Once a response generation is triggered, every data-point obtained from the start of the response is also saved ($\mathbf{Pa}_i$ and $\mathbf{P\dot{e}}_i$) for the duration of the response. To compute the error-rate, $\dot{e}_k$, the available (noisy) reconstruction errors, $\mathbf{e}$, are first passed through a smoothing filter, f_s, to compute the smoothed errors, $\widetilde{\mathbf{e}}$. The last two smoothed error values can then be used to compute the rate, $\dot{e}_k$, as:

$$\dot{e}_k = \frac{f_s(k) - f_s(k-1)}{\delta i} = \widetilde{e}_k - \widetilde{e}_{k-1}. \tag{1}$$

BO then proceeds to construct a model of the unknown relationship, $\dot{\mathbf{e}} = f(\mathbf{a})$, between error-rates, $\dot{\mathbf{e}}$, and actions, $\mathbf{a}$, for the given emergency scenario using GP regression. This GP model, $G[f(\mathbf{a})]$, is used to conduct the relatively simpler acquisition-function optimization to find the next best action, a_i^*, to take.

The optimal action, a_i^*, is then applied to the environment. At the subsequent time-step, the resulting error e_i will be obtained, from which $\dot{e}_i$ can be computed. Both $\mathbf{Pa}_i$ and $\mathbf{P\dot{e}}_i$ are then updated accordingly and the above BO loop procedure is repeated, until the response length, N, is reached.

4.1 Acquisition Function

We employ the Upper Confidence Bound (UCB) acquisition function ([5]), given by Eq. 2. Here, $\mu(\mathbf{a})$ and $\sigma^2(\mathbf{a})$ are the mean and variance of the regression model for the relationship, $\dot{\mathbf{e}} = f(\mathbf{a})$.

$$UCB = \alpha(\mathbf{a}, \beta, G[f(\mathbf{a})]) = \mu(\mathbf{a}) + \sqrt{\beta \cdot \sigma^2(\mathbf{a})}. \tag{2}$$

UCB is chosen since it includes a parameter, β, that allows direct control over the balance between exploration and exploitation, that is, how much the system should try actions that are far from those already sampled versus how much should it focus on the most promising actions found so far. It can be effectively optimized using quasi-Newton methods such as the L-BFGS-B algorithm [14].

4.2 Exploration/Exploitation Trade-off

Since an emergency response is time-critical, it is important to ensure a transition from an initial exploratory behavior to an exploitative one in a timely manner so that the search converges on an effective solution fast enough to avoid the danger. To accomplish this, the explicit parameter, β, is set to a decreasing function of time, $\beta(t_i)$, $i \in [1, N]$. The initial value, β_0, must be relatively high to encourage the BO to explore the action-space. As the action-sequence progresses, this parameter should decrease to a relatively lower value, β_k, so that the optimization begins to exploit the best solution found so far. These requirements produce the following constraints on the form of the time-varying function chosen for $\beta(t_i)$:

$$\beta(t_1) = \beta_0, \tag{3}$$

$$\beta(t_i \geq t_k) = \beta_k, 1 < k \leq N, \tag{4}$$

$$\beta_0 > \beta_k, \tag{5}$$

$$\frac{d\beta(t_i)}{dt_i} \leq 0, \quad \forall t, \quad t_1 \leq t \leq t_N. \tag{6}$$

In this way, the degree of initial exploration by BO can be controlled by the choice for β_0, and the degree to which it exploits the best solution found so far can be controlled through the choice for β_k.

4.3 Temporal Relevance of Data

A second point of concern in devising the acquisition function is incorporating the influence of time. The underlying relationship between error-rate and actions would, in general, be time-varying. Thus, recent observations will have greater relevance to, and influence on, the decision being made at any given time-step compared to older observations. To account for this temporal variation, we propose a penalty function that discounts the utility of any given observation, based on that observation's "age" within the time-span of the response action-sequence.

The utility of any given action is given by the UCB acquisition function, which depends on the GP model used to obtain $\mu(\mathbf{a})$ and $\sigma^2(\mathbf{a})$ (see Eq. 2). The GP regression model captures the influence of past observations on any other unseen one being estimated based on their relative distances in action-space. Thus, the discounting of action utility must be incorporated into the error-rate data used to compute the GP model. As such, we define a penalty function that operates directly on the set of error-rates available at any given time-step of the response. In particular, at the i^{th} time-step of a response action-sequence, each error-rate, $\dot{e}_j$, in $\mathbf{P}\dot{\mathbf{e}}_i$ is transformed to a discounted measure, $\hat{\dot{e}}_j$, through a penalty function, $q(\dot{e}_j, \tau_j)$, before computing the GP regression, where:

$$\tau_j = i - j, \quad \forall j, \quad 1 \leq j \leq (i-1), \text{and} \tag{7}$$

$$\frac{dq}{d\tau} \geq 0, \quad \forall \tau \geq 1. \tag{8}$$

Here, τ_j represents the age of the j^{th} error-rate at time-step i, and Eq. 8 indicates that the penalties should increase with age. This user-specified penalty function can be devised under this constraint depending on how strongly and quickly one wishes past data to lose its significance. An example is provided in the experiments presented in Section 5.

5 Simulation Experiments

To demonstrate and validate the proposed method, experiments with two different types of emergency situations were conducted using the open-source CARLA autonomous driving car simulator [9]. In the first situation, the A-EMS method was used to safeguard an agent from unexpected lane-drifting that it has not been designed to detect and correct-for. In the second emergency situation, the proposed response-generation method was used to detect and avoid imminent collisions with obstacles that an agent has never encountered before.

Each simulation run proceeds in discrete time-steps. At the start of each time-step the agent receives an observation corresponding to the current system state in the form of an RGB image from its forward-facing camera sensor. The agent then selects an action, namely, the throttle, brake, and steering inputs. The simulation updates the system state accordingly to the next time-step using the selected action. This process repeats until the end of the simulation run.

The VAE used to compute reconstruction errors is trained offline on images gathered from observations made under nominal conditions. Here, the agent is controlled via the built-in CARLA auto-pilot and made to drive on the same sections of road used in the experiments, but without introducing any emergency situation. A total of 72000 images obtained this way were used to train the VAE.

5.1 Lane-Drifting Experiments

Experimental Setup In these experiments, a gradual drift to one side is induced in the autonomous car as it drives along a straight section of road simulated in CARLA. To ensure no reliance on, or influence from, the agent's policy or learning mechanism on the response generation, the agent was controlled by the built-in auto-pilot software in CARLA, modified to enable only straight driving in the left-hand lane with no action taken to correct drift.

After a period of driving straight unfettered, the steering control inputs are altered so as to cause the car to begin drifting into the right-hand lane. As a result, the incoming observations gradually change to those that are unexpected compared to the nominal driving that the agent has experienced before. An RL agent not trained to deal with such inputs could collide with another vehicle in the right-hand lane, or even drift off the roadway, as it tries to learn the optimal response through its trial-and-error process.

The proposed A-EMS method is triggered to generate a corrective response to curtail this drift with neither any prior experience or training in doing so nor any context-specific information to indicate what exactly the problem is. The response generated by the A-EMS method is used to take-over the actions output by the agent's existing policy over the next N=80 time-steps.

As we are unaware of existing emergency-response methods for novel situations where no prior training or preparation is employed for the scenarios encountered, A-EMS is compared with a random response, where actions are selected at random at each of the next N=80 time-steps. For a fair comparison, both the A-EMS and the random responses are artificially triggered at the same

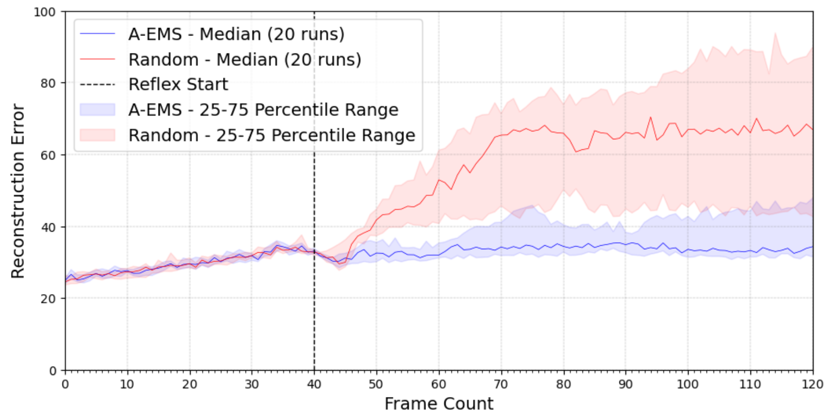

Fig. 2: Comparison of VAE reconstruction errors between the A-EMS and random-response approaches over all experiment runs, from initiation of the drift to completion of the N=80 time-step response. Shaded regions represent the 25$^{\text{th}}$ to 75$^{\text{th}}$ percentile ranges of the errors over all runs at each time-step.

point in time in their respective simulation runs after the drift is initiated. In all experiments, the vehicle maintained an average forward speed of 20km/h prior to the emergency response. Equations 9 and 10 give the functions used for the trade-off parameter, β, and the time-based penalties on the error data, respectively. These functions were used for illustrative purposes and the user is free to design them as they see fit under the restrictions stipulated by Equations 3 - 8.

$$\beta(ti) = -0.0028t_i^2 + 0.07, \quad i \in [0, N-1], \tag{9}$$

$$q(\dot{e}_j, \tau_j) = 0.02269e^{(0.2293\tau_j)}, \quad i \in [1, N-1], j \in [0, i-1]. \tag{10}$$

Results A total of 20 runs of the lane-drifting experiments were conducted under each of the response-generation approaches: A-EMS and random-response. Figure 2 shows plots of the VAE reconstruction errors that resulted from all runs for both these methods. Figure 3 shows plots of the 2D position coordinates of the centre-of-mass of the vehicle recorded over all experiment runs, from initiation of the drift to the end of the N = 80 time-step response.

Responses that maintained a lateral center-of-mass deviation below 1.5 m relative to the center of the left-lane were considered to be successful. The experimental results showed a 8/20 = 40% success-rate for the random-response approach and a 14/20 = 70% success-rate under the proposed A-EMS method.

5.2 Collision-Avoidance Experiments

Experimental Setup Nine different collision scenarios were setup in CARLA within a simulated urban environment (see Figure 4), each involving a different, unforeseen, stationary obstacle placed in the path of the autonomous car

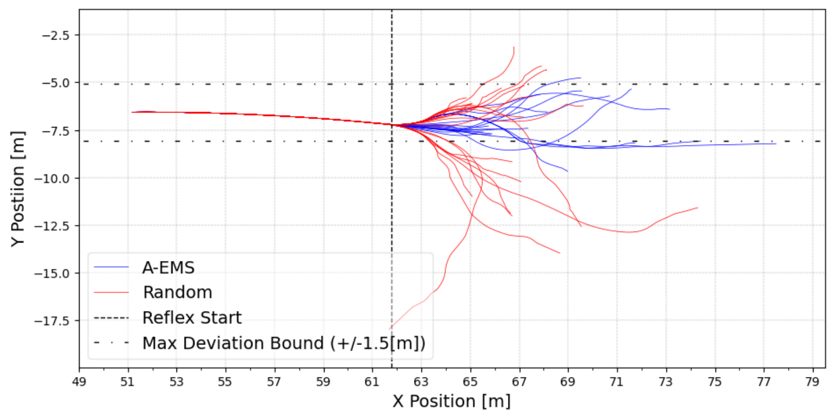

Fig. 3: Comparison of paths traced out by agent over all runs between the A-EMS and random-response approaches.

driving along a section of road in one of 5 different parts of the map. These scenarios simulate a situation where an autonomous driving agent, assumed to have been trained to drive in an obstacle-free urban environment, is suddenly presented with an unforeseen situation involving a stationary obstacle placed in its path. The trial-and-error learning process for an RL agent in such a situation could involve taking dangerous actions, possibly resulting in collisions with the obstacles.

Two sets of experiments were conducted within this emergency situation. In the first, the CARLA auto-pilot maintained an average agent speed of 20 km/h before encountering an obstacle, and the A-EMS method was combined with an example emergency-detection method. In the second set, an average speed of 30 km/h was maintained and response-generation was artificially triggered.

In the first set of experiments a rudimentary, VAE-error-based emergency-detection method was used, only as an example to demonstrate how the A-EMS method could be combined with a detection algorithm to compose a complete, independent, monitoring module that takes over the agent's output with an emergency response only when necessary. This emergency-detection mechanism uses a straightforward approach similar to that presented in [15]. In particular, at each time-step, a second-order polynomial regression fit is computed on the last M=15 VAE reconstruction errors, which are then extrapolated K=7-time-steps into the future and compared against an upper-bound threshold, ULe, to indicate the presence of a novel, unforeseen situation requiring an emergency response. While any mechanism suitable to the application being considered can be used to identify when a dangerous situation is imminent, an auto-encoder-based approach presents a natural choice given that the response-generation method employs a VAE as a key part of its input sensor-data processing.

The emergency-detection component monitors the actions of the agent and the observations received from its camera sensor. Upon detecting an imminent

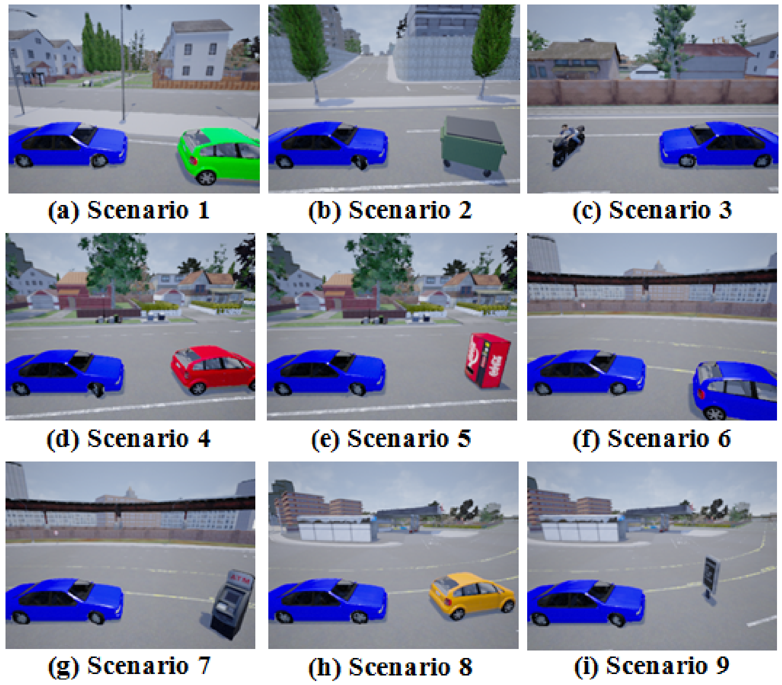

Fig. 4: Simulated stationary-obstacle scenarios used in the collision-avoidance experiments: (a) location A, green car; (b) location B, garbage container; (c) location B, motorcycle; (d) location C, red car; (e) location C, vending machine; (f) location D, blue car; (g) location D, ATM machine; (h) location E, orange car; (i) location E, street-sign.

collision as the agent approaches a stationary obstacle, the module triggers the A-EMS algorithm to takeover the agent's actions for the pre-specified next N=30 time-steps with a customized action-sequence to attempt to prevent this collision.

In the second set of experiments the emergency-detection mechanism is replaced by an artificial trigger that initiates the different response-generation approaches being compared at the same point in time during each simulation run. Without the additional time-delay caused by a separate emergency-detection component, the simulations could be run at a faster average forward agent speed of 30 km/h. In both sets of experiments a worst-case scenario is simulated where, in the absence of the emergency-response system, the agent takes no action to avoid the obstacle and continues to follow the road.

Results In the first set of experiments the A-EMS method was compared with a random-response approach. Twenty repetitions for each scenario were conducted and the percent of successful collision-avoidance runs (i.e., success-rate) was computed. Table 1 summarizes the results.

Table 1: Summary of success-rates for simulated collision-avoidance scenarios (note: taking no action resulted in a 0% success-rate in all cases).

Scenario	**Exp. Set 1 (20 km/h tests)**		**Exp. Set 2 (30 km/h tests)**	
#	**A-EMS**	**Random**	**A-EMS**	**Random**
1	80%	10%	82%	5%
2	65%	20%	90%	25%
3	75%	35%	75%	10%
4	75%	50%	80%	25%
5	75%	25%	85%	70%
6	45%	5%	55%	40%
7	75%	30%	70%	25%
8	70%	25%	25%	7.5%
9	80%	15%	57.5%	17.5%
Avg.:	**71%**	**24%**	**68.8%**	**25%**

In the second set of experiments, both the proposed approach and the random-selection approach were triggered manually at the same time for all scenarios. This ensured that the same distance and initial approach speed existed for the approaches compared (see Table 1 for success-rate results). As a representative example for illustration, Figure 5a shows a plot of the variations in VAE reconstruction errors, and Figure 5b gives a closer look at the error-rates themselves, over the span of the response action-sequences for Scenario 1 in the second set of experiments where the alternative responses are triggered at the same time for a fair comparison. For reference, Figure 5 also includes the errors and rates that result from taking no action upon encountering the obstacles.

6 Conclusions and Discussion

This paper proposed A-EMS: an emergency-response method that enables an autonomous lifelong-learning agent to safely address unforeseen situations encountered during deployment for which the existing policy becomes unreliable. When triggered, the method generates a response by finding optimal actions sequentially through minimization of VAE reconstruction error rates from the novel observations using a modified BO algorithm. Simulation experiments in an autonomous car-driving domain demonstrate how minimization of observation uncertainty using A-EMS can find safe actions to curtail unexpected lane-drifts and also to avoid collisions with never-before-seen obstacles, despite never having encountered such scenarios before.

The significantly greater average success-rate by A-EMS in controlling lateral drift and in collision-avoidance compared to a random approach indicate that effective, intelligent actions are indeed being selected to avoid the novel dangerous situations, beyond simply what random chance would allow. This demonstrates how minimizing a measure of uncertainty in the observations can be correlated

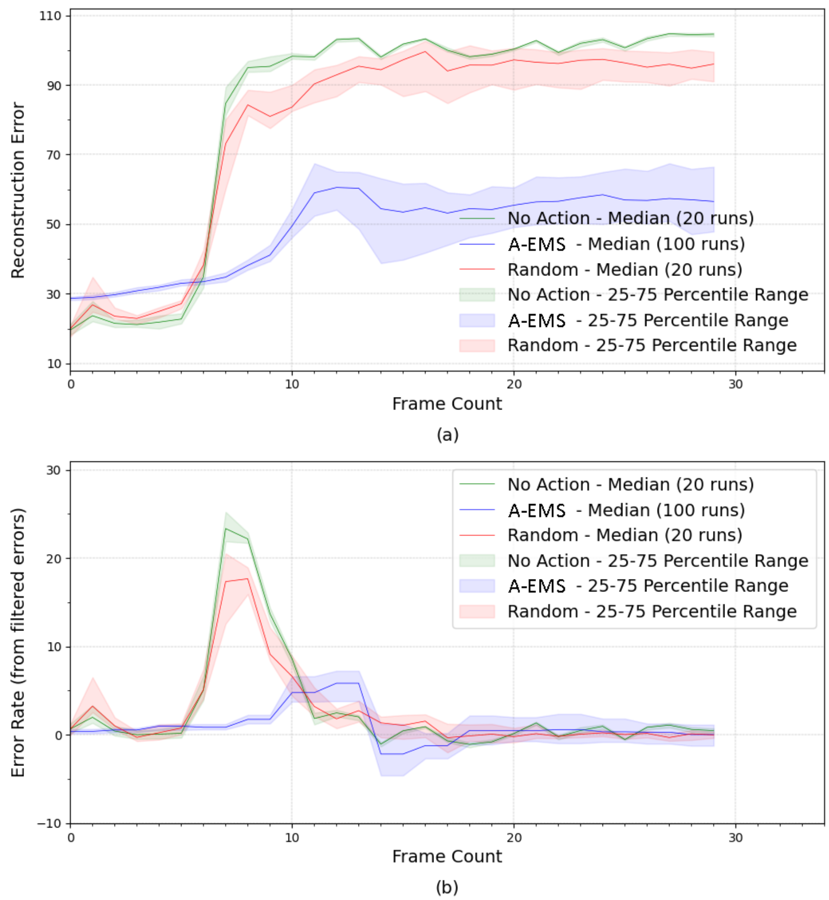

Fig. 5: Comparison of the impact of A-EMS, a random response, and no-action, on (a) VAE reconstruction errors, and (b) VAE reconstruction error-rates, over the Scenario 1 runs in Experiment set 2.

with good actions that help to effectively deal with unforeseen situations. These effective, danger-avoiding behaviors are also reflected in the reconstruction errors themselves (Figures 2 and 5a), where the errors rise relatively slowly and plateau at a relatively lower final value due to the agent having transitioned to a more familiar state.

Some scenarios from the second emergency situation simulated presented more of a challenge than others. Detection of the imminent collisions in experiment-set 1 was observed to happen when the agent was on average about 8 m away from the obstacle. This left little distance and time to react, and in some cases it was not enough for the final response to avoid the collision, even though it may have been effective had the danger been detected sooner. Qualitative observations of some of the failures under the proposed method show that the agent still tries to make sensible maneuvers to avoid the collision and almost succeeds.

It should be noted that the intention here was not to create the best drift-correction or obstacle-avoidance system for an autonomous car, but rather to demonstrate how minimization of observation uncertainty can be an effective driver to safely address novel situations for which a learning agent would otherwise be unprepared.

Moreover, A-EMS does not require context-specific information either (i.e., understand the significance of lane-markings or know what the obstacle is, or what its presence means in the context of driving). As such, the performance of the method can always be improved by incorporating context-specific mechanisms on top of the basic emergency-response system for the particular application being addressed, if so desired.

References

1. Achiam, J., Amodei, D.: Benchmarking safe exploration in deep reinforcement learning. https://d4mucfpksywv.cloudfront.net/safexp-short.pdf (2019), in NeurIPS Deep Reinforcement Learning Workshop
2. Achiam, J., Held, D., Tamar, A., Abbeel, P.: Constrained Policy Optimization. In: Proceedings of the 34th International Conference on Machine Learning - Volume 70. p. 22–31. JMLR.org, Cambridge, MA, USA (2017)
3. Arulkumaran, K., Deisenroth, M.P., Brundage, M., Bharath, A.A.: Deep Reinforcement Learning: A Brief Survey. IEEE Signal Processing Magazine **34**(6), 26–38 (2017)
4. Bengio, S., Vinyals, O., Jaitly, N., Shazeer, N.: Scheduled Sampling for Sequence Prediction with Recurrent Neural Networks. In: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1. p. 1171–1179. MIT Press, Cambridge, MA, USA (2015)
5. Brochu, E., Cora, M., de Freitas, N.: A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning. Technical Report TR-2009-023, Department of Computer Science, University of British Columbia (2010)
6. Brown, N., Sandholm, T.: Libratus: The Superhuman AI for No-Limit Poker. In: Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence (IJCAI-17). pp. 5226–5228. IJCAI Organization, Menlo Park, Calif (2017)
7. Caselles-Dupré, H., Garcia-Ortiz, M., Filliat, D.: S-trigger: Continual state representation learning via self-triggered generative replay. International Joint Conference on Neural Networks (IJCNN 2021) (2021), accepted
8. Deng, Y., Bao, F., Kong, Y., Ren, Z., Dai, Q.: Deep Direct Reinforcement Learning for Financial Signal Representation and Trading. IEEE Transactions on Neural Networks and Learning Systems **28**(3), 653–664 (2017)
9. Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., Koltun, V.: CARLA: An Open Urban Driving Simulator. In: Proceedings of the 1st Annual Conference on Robot Learning. p. 3521–3526. PMLR, Bletchley Park, UK (2017)
10. Fisac, J.F., Akametalu, A.K., Zeilinger, M.N., Kaynama, S., Gillula, J., Tomlin, C.J.: A General Safety Framework for Learning-Based Control in Uncertain Robotic Systems. IEEE Transactions on Automatic Control **64**(7), 2737–2752 (2019)

11. Francois-Lavet, V., Henderson, P., Islam, R., Bellemare, M., Pineau, J.: An Introduction to Deep Reinforcement Learning. IEEE Signal Processing Magazine **11**(3-4), 219–354 (2018)
12. Ha, D., Schmidhuber, J.: World models (2018)
13. Kingma, D.P., Welling, M.: Auto-encoding variational bayes (2014)
14. Liu, D., Nocedal, J.: On the limited memory BFGS method for large scale optimization. Mathematical Programming **45**(1), 503–528 (1989)
15. Manevitz, L., Yousef, M.: One-class document classification via Neural Networks. Neurocomputing **70**(7), 1466–1481 (2007)
16. Marchi, E., Vesperini, F., Eyben, F., Squartini, S., Schuller, B.: A novel approach for automatic acoustic novelty detection using a denoising autoencoder with bidirectional LSTM neural networks. In: 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 1996–2000. IEEE Press, Piscataway, New Jersey, USA (2015)
17. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., Hassabis, D.: Human-level control through deep reinforcement learning. Nature **518**(7540), 529–533 (2015)
18. Mockus, J. (ed.): Bayesian Approach to Global Optimization: Theory and Applications. Kluwer Academic Publishers, Boston, MA, USA (2013)
19. Nguyen, C.V., Li, Y., Bui, T.D., Turner, R.E.: Variational Continual Learning. In: Sixth International Conference on Learning Representations. pp. 1–18. iclr.cc, La Jolla, CA, USA (2018)
20. Pan, X., You, Y., Wang, Z., Lu, C.: Virtual to Real Reinforcement Learning for Autonomous Driving. In: Proceedings of the British Machine Vision Conference (BMVC). pp. 11.1–11.13. BMVA Press, London, UK (2017)
21. Peng, X.B., Berseth, G., Yin, K., Van De Panne, M.: DeepLoco: Dynamic Locomotion Skills Using Hierarchical Deep Reinforcement Learning. ACM Transactions on Graphics **36**(4), 1–13 (2017)
22. Rasmussen, C.E., Williams, C.K.I. (eds.): Gaussian Processes for Machine Learning. MIT Press, Cambridge, MA, USA (2006)
23. Richter, C., Roy, N.: Safe Visual Navigation via Deep Learning and Novelty Detection. In: Proceedings of Robotics: Science and Systems. pp. 1–9. MIT Press, Cambridge, MA, USA (2017)
24. Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., Fergus, R.: Intriguing Properties of Neural Networks. In: 2nd International Conference on Learning Representations, ICLR 2014. pp. 1–10. iclr.cc, La Jolla, CA, USA (2014)
25. Tessler, C., Mankowitz, D.J., Mannor, S.: Reward constrained policy optimization (2018)
26. Thananjeyan, B., Balakrishna, A., Nair, S., Luo, M., Srinivasan, K., Hwang, M., Gonzalez, J.E., Ibarz, J., Finn, C., Goldberg, K.: Recovery RL: Safe Reinforcement Learning With Learned Recovery Zones. IEEE Robotics and Automation Letters **6**(3), 4915–4922 (2021)
27. Thananjeyan, B., Balakrishna, A., Rosolia, U., Li, F., McAllister, R., Gonzalez, J.E., Levine, S., Borrelli, F., Goldberg, K.: Safety Augmented Value Estimation From Demonstrations (SAVED): Safe Deep Model-Based RL for Sparse Cost Robotic Tasks. IEEE Robotics and Automation Letters **5**(2), 3612–3619 (2020)