iv4XR

Intelligent Verification / Validation
for Extended Reality Based Systems

#7

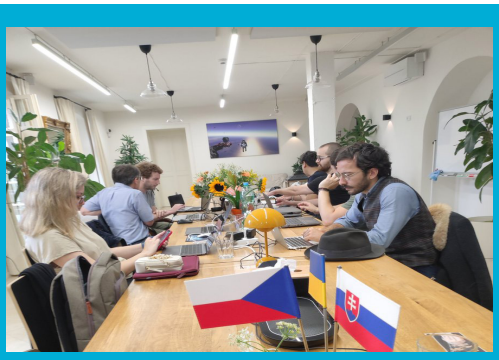
Project Newsletter
August 2022

/ About iv4XR

iv4XR - Intelligent Verification/Validation for Extended Reality Based Systems - is an H2020 European project focusing on the automated testing verification of extended reality (XR) systems through the use of autonomous and intelligent test agents. The project is in its second year and has so far made important progress in formalizing the problems and contextualizing them along the challenges faced by industrial partners. Solutions are being prototyped and applied gradually to the use cases.
For more information consult the project website

/ Project meetings

The seventh consortium meeting of the project was held in person on 29th June – 1st July, 2022 in Prague. This meeting was important for the consortium for discussing the plans regarding WorkPackages (WPs), final integration of the framework and validation of the results as the project is in the final year.



During the meeting, a detail session is held for iv4XR WPs where achieved milestones and ongoing activities are reported and discussed. Open issues within WPs are identified during this session and were discussed in separate working groups with the aim to identify potential solutions and collaboration.

A dedicated session was also held to discuss the integration of iv4XR framework and toolkit definition. The dissemination and exploitation activities of iv4XR are also reported which includes dissemination, communication & result management, exploration and valorization, and discussion of market research done by the industrial partners.

Doctoral students working on problems related to different aspects of verification/testing of XR based systems presented their ongoing work in a PhD symposium held prior to the consortium meeting.

The members of the consortium got the chance to meet the testers of GoodAI where the testers presented their current approach of testing and the difficulties they are facing. A vibrant session was held with the testers to discuss about the potential and challenges of an automated testing framework like iv4XR in facilitating their testing process by replacing the complex and cumbersome and time consuming current manual or scripted approach.


Autonomous agents to accelerate extended reality testing

/ Publications

We have so far managed to formulate the ideas and proposal of iv4XR and present the ongoing research and preliminary results obtained in various venues to get early feedback from the community.
Here are some of these articles recently published:

- A talk from the project co-ordinator Rui Filipe Fernandes Prada in the EU Research paper about the iv4XR project in developing autonomous testing agents that promise significant benefits to developers of Extended (XR) reality systems
- State Model Inference Through the GUI Using Run-Time Test Generation @RCIS'2022
- Scriptless Testing for Extended Reality Systems @ RCIS'2021

For details, check out our:
website: https://iv4xr-project.eu/publications/
Zenodo: https://zenodo.org/communities/iv4xr-project/

TESTAR at iv4XR

TESTAR is a tool that implements a scriptless approach for completely automated test generation for event-based Systems Under Test (SUT). Once the tool has sufficient information about the characteristics of the states of the SUT and what actions or events the SUT expects in a specific state, it can test the SUT fully automatically, without the use of programmed scripts. This is due to the agents that implement various action selection mechanisms and test oracles. The underlying principles are very simple: generate test sequences of (state,action)-pairs by starting up the SUT in its initial state and continuously selecting an action to bring the SUT into another state.
An integration has already been developed with the iv4xr Framework for LabRecruits and Space Engineers games, which allows the TESTAR tool to extract XR entities information, create an observable state that contains the properties of existing virtual entities and an additional navigable state that contains which were the reachable entities of the agent with the intention to execute more intelligent navigate-to-entity actions. Due to the complexity of the Space Engineers game, FBK and UPV are researching on how to measure coverage for this game.
TESTAR prioritizes the exploration of the iv4XR systems by navigating to unexplored positions to discover the reachable entities. In order to speed up the exploration process, we are investigating the implementation of a distributed architecture to execute multiple instances of TESTAR that use the state model as a central knowledge database. Currently, we have implemented and tested the first version of a distributed approach to apply TESTAR to web applications. We will continue this research applying this implementation to XR systems, such as LabRecruits and SE systems.
https://github.com/iv4xr-project/TESTAR_iv4xr

Model-based testing

One of the lines being pursued in iv4XR is the use of models to capture the desired behavior of the system under test (e.g., a game) in order to apply testing techniques based on the model. Our approach uses extended finite state machines (EFSMs) to capture an abstraction of the desired SUT behavior and search-based algorithms are used to derive abstract tests from the model, which are then concretized into action sequences that are executed on the game under test. Prototype of the tool is available in the project Github repository, It includes EFSM models of scenarios from LabRecruits as well as a generator for complex maze-like levels for LabRecruits. Different search algorithms are implemented for the generation of tests from the EFSM models. Furthermore, concretising and executing of abstract tests is also supported for LabRecruits, together with mutation analysis. These features are easily extensible to other SUTs.
The latest version includes implementation of the K-Transition coverage criterion, besides the existing state and transition coverage criteria.
https://github.com/iv4xr-project/iv4xr-mbt

Reinforcement Learning

For different aspects of the project, we are exploring the application of reinforcement learning (RL). In particular, we are exploring RL for:

- Testing the system under test (SUT) to achieve the exploration of different aspects of the behavior of the SUT (WorkPackage 3)

  > Different Reinforcement Learning strategies are being investigated. In particular, with different algorithms for the reward calculation in TESTAR, which consist on rewarding the actions that have not been executed a lot, rewarding the state changes, rewarding the changes in the widget tree, rewarding the changes on images by comparing pixels, and reward image changes by comparing similarity matrices. Results indicate that with rewarding algorithms the coverage is reached faster than random action selection mechanisms. After the proper evaluation of the rewards with big systems, the framework will be tested with XR systems.

  > We are focusing on defining a generic approach for providing coverage using RL solutions. To this end, we are investigating on formulating reasonable metrics for measuring coverage of the iv4XR pilots. In particular, we intend to use RL algorithms for automated play testing and providing functional coverage for iv4XR pilots. Use of RL solutions in this complex partially observable scenario is challenging. We have defined a curiosity driven reward based reinforcement learning approach that has the ability to become a powerful exploration mechanism to facilitate RL agent to explore the space of interactions in the game, hence increase the coverage. The reward function encourages the discovery of previously unseen states and discourages immobility and revisiting of already seen states. Results are promising where the curiosity-based RL is effective in achieving reasonable levels of coverage, in particular on larger and complex game scenarios/levels.

  > One of the pilots of iv4XR framework is the verification of the defense strategy of a critical infrastructure against an infiltration. In such a scenario, Deep RL approaches are being investigated to aid the adversarial testing where the testing agents try to defeat the defense strategy of a nuclear plant infrastructure. In this context Diversity RL is used where the main idea is is to use a DRL solver to achieve behavioral coverage. Whereas in a classic RL training setup a single control policy is learned that fulfills the goal, Diversity RL allows the learning of a set of diverse and successful policies to fulfill an identical goal. In this pilot focusing on the defense mechanism of a critical infrastructure, Diversity RL allows to obtain different intrusion strategies due to different flaws of the defense strategy, that can thus be corrected by the SUT user.

Initial investigation and implementation has successfully finished. Fine-tuning of the algorithm is carried out with the adaptation of QD-RL. Fine-tuning of the comparison metric between two RL trajectories is also performed. Initial experiments are completed in the toy maze environment, with up to 2 guards and 5 different starting positions for the intruder. Experiments going on with increasing the problem complexity of the environment.

- Exploring different behavioral aspects and dimensions of the affective perspective related to XR based systems. This includes, but not limited to, exploring collaborative behaviors among test agents (WorkPackage 4)

Augmented Reality Testing

Taking as a reference the Google ARCore project, capable of creating Augmented Reality experiences, we want to implement tests that evaluate properties such as the position and size of AR objects in AR environments. We are starting the research in this line by adding Record and Playback functionalities in order to allow to record an AR session and run tests directly on the recording.

Automated UX testing

The project is exploring the use of agents endowed with affective and cognitive models to automatically assess User eXperience (UX). The objective is to develop socio-emotional test agents (SETAs) to aid the systematic assessment of user experience of XR systems while minimizing the manual effort.
We aim to create a toolset that allows developers to chose the UX metrics that are more relevant for their product and we are currently developing a framework for automated testing of user experience that integrates the work we have been developing.
We have work on emotional and cognitive models, automatic assessment of a game level difficulty, narrative paths (for interactive stories) and personas (agents that simulate the behaviour and preferences of different types of player during gameplay).

Multi-Agent testing

Many XR systems allow the simultaneous interaction of multiple users in the same environment. This implies the need to verify the correct interaction of multiple users, since they can influence each other. Various activities are ongoing focusing on extending the iv4XR framework to allow the communication of multiple agents in runtime. These activities concentrate on the following two objectives : (1) allow the definition of test cases that involve simultaneous interactions, collaboration or confrontation of multiple agents, and, (2) improve entity-search and exploration performance by coordinating a group of agents to achieve a common goal. In particular, use of multi-agent approach in testing and providing coverage for iv4xr pilots, where multiple agents collaborate to speed up the search for solution(s) or to do testing.

Such modeling is not trivial, it calls for special consideration for defining the sharing and coordination of knowledge between intelligent agents in an distributed approach. Specially, the problem becomes more complex in a dynamic environment like iv4xr pilots where it requires multiple sequential decisions and the agents typically have only limited information about the system.

Integration of use cases into iv4XR framework

One of the objectives of the iv4XR project is to encourage external organizations to use the framework to test and monitor their extended reality environments with less human interaction than is required by the testing methods of today. The pilots are one of the methods that the consortium is using to demonstrate the benefits of using iv4XR and how to integrate the framework into their development lifecycle. There are three pilots which are in the phase of full integration with the iv4XR testing framework. Full integration concentrates on a "feature complete" version of the interfaces so that the developer of a test agent has access to all of the functionality and internal information required in order to test the salient features of pilots.

For the full integration, the functionality of the interface has been expanded to the accessibility of the mutable and static properties of any block in the game. The academic consortium partners have been making use of the interface for their experiments and have been constructing agents to interact with SE. This has been a way to generate feedback for the interface and for ideas to improve the workflows of those that create agents.
https://github.com/iv4xr-project/iv4xrDemo-space-engineers

Similarly, the pilot from Thales on intrusion detection has been integrated into the iv4XR platform. The prototype implementation of the integration allows some basic commands to be exchanged between iv4XR and the pilot application. It is available in the project Github repository. The objective of the "Full Integration" phase is to improve the pilot on two aspects : (i) accelerating the simulation in order to allow the use of Reinforcement Learning; (ii) expansion of the capacities of the interface in order to allow the AI tools to access more simulation data or parameters. This integration aims to fulfill all the requirements needed to allow an external AI tool, such as Thales SIX Reinforcement Learning (RL) algorithms, to challenge the defense strategy implemented in MAEV. To achieve this objective, the CGE should be able to run the simulation much quicker than real time in order for the RL algorithms to test and evaluate thousands of alternatives as quickly as possible. The capacities of the interface have been expanded in order for the AI tools, not only to control MAEV agents, but also to control the course of the simulation and to access the simulation data that are needed to evaluate the alternatives.
https://github.com/iv4xr-project/iv4XR-IntrusionSimulation

For the LiveSite pilot, a server-side tool is developed which can interface with the iv4XR framework. Its inputs are monitoring projects with sensor definitions, thresholds, and their varying requirements, and it uses the IV4XR framework to test parameters within the definition of the given sensors. For the intermediate integration phase, the objective has been to further enhance this tool to allow both processing and navigation of the project, by allowing the tool to control which sections of the data it is looking at. In effect, it is an agent navigating through the data. For full integration the system is advanced to analyze the formulae for inter-dependent sensors which are frequently found on large structures such as bridges and buildings.

/ Check out our channels

We have set up various channels where we regularly disseminate updates and progress on our project. Follow us on your preferred channel:

Twitter: https://twitter.com/iv4xr
Facebook: https://www.facebook.com/iv4xr
LinkedIn: https://www.linkedin.com/company/iv4xr-project
GitHub: https://github.com/iv4xr-project
Zenodo: https://zenodo.org/communities/iv4xr-project

H2020 RIA project funded by the EU under grant number: 856716
Total funding: €3 725 337.50