POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

Executive Summary of the Thesis

Byzantine Fault-Tolerant Swarm-SLAM through Blockchain-based Smart Contracts

Laurea Magistrale in Automation and control Engineering - Ingegneria dell'Automazione

Author: Angelo Moroncelli

Advisor: Prof. Francesco Amigoni

Co-advisor: Dr. Andreagiovanni Reina

Academic year: 2023–2024

1. Introduction

Autonomous robotic systems face the challenge of navigating unknown environments without relying on external localisation systems. To address this, Simultaneous Localisation And Mapping (SLAM) becomes crucial. Through SLAM techniques, robots can create real-time maps and determine their positions within the environment.

While single-robot SLAM has seen efficient solutions from decade of research, recent focus has shifted to the development of multi-robot SLAM systems. These systems involve groups of robots collaborating to build maps and enhancing localisation through information exchange. Multi-robot SLAM offers opportunities for increased efficiency, robustness and parallelisation. Moreover, parallel to the evolution of SLAM technology, swarm robotics has proven to be promising for exploring vast environments with fully autonomous decentralised robots, all while maintaining adaptable properties.

Swarm robotics is an innovative approach to robotics inspired by the collective behaviour of social animals. By following simple rules and engaging in short-range interactions, swarm robotics aims to design and create systems composed of a large number of robots that self-organise to perform tasks cooperatively [1].

Considering SLAM and swarm robotics together has become natural to exploit the advantages of robot swarms in self-organising and redundant systems comprising a potentially high number of entities, that are performing simultaneous localisation and mapping. In 2021, Kegeleirs et al. reviewed the state of the art in SLAM with robot swarms, a rather novel approach that in some way extends what is done in the multi-robot applications [2]. SLAM with robot swarms brings challenges related to scalability and consistent information aggregation, especially when dealing with potentially conflicting data. Additionally, although robustness is often indicated as an intrinsic characteristic of multi-robot systems thanks to the presence of many robots, recent research [5] [4] has shown that robot redundancy and parallelisation of operations are not sufficient to achieve system robustness against misbehaving robots. It is reasonable to assume that a subset of robots may misbehave, deviating from the designed algorithm, due to internal errors or to external malicious tampering. In agreement with decentralised system literature, I name such misbehaving robots as Byzantine robots.

This thesis studies the robustness of the state-of-the-art framework for multi-robot SLAM in decentralised robot swarms, which is called Swarm-SLAM [3]. I show that Swarm-SLAM is vulnerable to the presence of Byzantine robots: even one of them is sufficient to jeopardise the entire system. Therefore, inspired by recent research on blockchain-based swarm robotics [5] [4], I built a security layer for Swarm-SLAM through blockchain-based smart contracts, which are distributed algorithms running on the blockchain. I test my solution with a set of physics-driven simulations of groups of eight robots running algorithms coded in ROS2 and a custom blockchain framework. The results show that the proposed blockchain-based solution makes Swarm-SLAM tolerant to relatively large Byzantine faults.

My analysis also shows that the increase in Byzantine fault tolerance is compensated by a decrease in system efficiency. This thesis discusses such a robustness-efficiency trade-off and also comprehensively discusses the security issues that Swarm-SLAM, in the specific, and multi-robot SLAM, in general, face and how they could be potentially addressed through future research in blockchain-based solutions for robotics.

2. Motivations

In single-robot SLAM, the robot simultaneously estimates its location and constructs a map of its surroundings. The main issue is that the error in the robot's position can accumulate over time, leading to inaccurate maps. A loop closure, in this context, refers to the process of detecting and incorporating information about the robot's revisit of a previously visited location within its environment. Intra-robot loop closures are fundamental for improving the accuracy and consistency of a SLAM outcome.

In a multi-robot SLAM scenario, things are more complex due to the presence of multiple robots that are collectively exchanging information with the goal of accelerating convergence towards a unified map thanks to collaboration. Specifically, they are able to create inter-robot loop closures, that are a multi-agent extension of the intra-robot case, while the latter is still present. Clearly, inter-robot loop closures require information sharing among participants, and their creation needs that the same location is visited by multiple robots. In Swarm-SLAM, even the injection of a single incorrect loop closure in the pose graph is hazardous for the map generation and the simultaneous localisation of every robot. One incorrect loop closure can lead to a very high localisation and mapping inaccuracy.

Consequently, security vulnerabilities that permit information sharing from robots with faulty or malicious behaviours can pose significant risks. For instance, when robots are allowed to freely participate in a collective action and one of them behaves in a way that is not beneficial for the swarm, this leads to overall performance degradation and the swarm needs a mechanism to avoid it [5]. Indeed, preventing collaborative systems from false data injection and mitigating malicious attacks is a necessary step towards robotics applications with safety guarantees.

Although state-of-the-art multi-robot SLAM methods are called "robust", authors often mean robustness against outliers due to perceptual aliasing. They define an outlier as data that deviates significantly within a set of measures, but if the measures are sparse and very different in magnitude from each other they are hardly identified and rejected. I demonstrate that Swarm-SLAM cannot cope with big outliers, or even with groups of wrong loop closures injected by Byzantine robots. Incorrect inter-robot loop closures are accepted as correct measures.

However, a reliable Swarm-SLAM system should be robust to incorrect inter-robot loop closure injection, but there is a notable absence of studies or implementations related to fault tolerance in SLAM with robot swarms. Protecting the system from the injection of incorrect loop closures by Byzantine robots is of central importance, since loop closures are fundamental for pose graph optimisation. Therefore, I want to ensure that Swarm-SLAM is secure against unauthorised inter-robot loop closures.

In Swarm-SLAM, a loop closure is generated when a robot (let's call it Robot i) recognises the same scene that another robot (Robot j) saw in the past. When they meet, Robot j sends its image frame relative to the corresponding scene to Robot i, and Robot i calculates the geometric loop closure. I found that a loop closure can have two main sources of error:

1. Wrong loop closure calculation from Robot i.
2. False information sent by Robot j or alteration of the information from the one that is receiving the data.

Currently, every agent could theoretically calculate false geometric transformations concerning the perceived data. Hence, they could create loop closures that work as wrong constraints in the pose graph optimisation, leading to incorrect results. Figure 1 illustrates a comparison between a correct process, where an exact inter-robot loop closure is calculated from Robot i to Robot j, and another scenario in which the sender of the loop closure is a Byzantine robot, resulting in a wrong outcome.

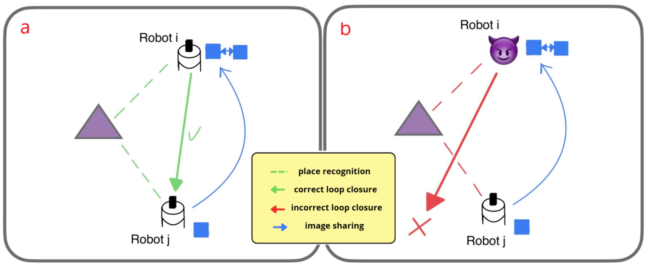

Figure 1: The dashed lines indicate that, in different moments, two robots acquired information about the same scene (illustrated through a purple triangle); which corresponds to two matching descriptors during the meeting. In this scenario, Robot j sends the image (blue square) that has the matching descriptor to Robot i.

In panel **a** is depicted the correct process that should take place in case of a rendezvous between two reliable robots. The green arrow represents a correct inter-robot loop closure from Robot i to Robot j.

Contrarily, in panel **b** is represented the construction of an incorrect loop closure. Robot i is a Byzantine robot (represented by the purple evil) and it generates an incorrect loop closure (red arrow) from Robot i to Robot j.

3. Approach

In this thesis I propose a Byzantine fault-tolerant protocol, that is a class of methods designed to handle failures, including malicious failures or arbitrary faults, in a distributed system. I employed blockchain technology to do that.

Blockchain technology is a revolutionising paradigm for distributed data management and secure information storage that allows users to exchange economic transactions without relying on a central authority. In a couple of words blockchain is a shared, tamper-proof and immutable ledger that facilitates the process of recording transactions and tracking assets in a network. Blockchain permits the use of decentralised computing platforms to control the injection of information in systems where direct verification among participants is not possible, preserving privacy and scalability. Moreover, blockchains can be used to manage robots' permissions inside a network, changing the robots' decision power in the swarm based on reputation [5].

Compared with a centralised controller, blockchain gives several advantages. Thanks to its decentralisation, it can be completely integrated within Swarm-SLAM. The single point of failure is removed. Indeed, blockchain distributes the control of information among the swarm, reducing the risk of having only one central server which would not be desirable in Swarm-SLAM. A blockchain also ensures that the data it stores is tamper-proof and thus cannot be changed by an ill-intent robot. Lastly, the possibility of having identical smart contracts—i.e., shared programs— that are guaranteed to be executed in the same way by everyone ensures trust among system users.

These are just few key factors that motivate me to propose the use of blockchain to improve the reliability of Swarm-SLAM.

I designed and implemented a blockchain-based smart contract, i.e., an algorithm running on the blockchain, that is able to check security requirements over the loop closures that are proposed by every robot in order to prevent the injection of permission-less information in the Swarm-SLAM back-end. This smart contract is based on geometric relationships that the correct transformations between robot poses have. It is able to reject incorrect loop closures, at the cost of introducing latency in the system.

Once a set of inter-robot loop closures are validated from the blockchain, they are automatically shared among the robots from the one that measured them, and they can be used to improve the map's accuracy. During the entire process, an inter-robot loop closure is treated as high-level and lightweight information that represents a geometric transformation between two robot poses, based on raw and heavy data, e.g., images or LiDAR point clouds. Such raw heavy data do not need to be further considered once the relative loop closures are created and incorporated into blockchain transactions, preserving scalability in swarm systems.

In Swarm-SLAM, the exchange of information must be minimal. During rendezvous, only one robot sends its useful data to the other, and not vice-versa, following local interaction rules. The receiver calculates the geometric transformation and only the loop closure, a sort of high-level data, is used for pose graph optimisation. If each robot sent its images to every encountered peer about the individuals encountered in the past for direct verification, the system would not be scalable. Instead, by collecting only the loop closures on the blockchain, we can save and exchange lightweight yet fully informative information in a secure way. Each robot keeps and exchanges its own perception data only. Moreover, blockchain's smart contract takes the role of a meta-controller that decides who is the robot entitled to perform some action, limiting the power of Byzantine robots.

4. Results

The major result I obtained in simulation through the application of my method can be summarised in the two following plots. Here, simulation results for different numbers of Byzantine agents are shown. The malicious or faulty behaviour is the addition of a constant 9.0 metres error on translations, at every loop closure calculation. As shown in Figure 2, in case of a non-secured systems, the error in the final 2D map tends to increase with the number of Byzantine robots, when the perturbation consists in constant noise injection. Clearly, this corresponds with the fact that: the more Byzantine robots there are, the more wrong loop closures are used. Contrarily, when Swarm-SLAM is secured by the blockchain-based smart contract the final pose graph fits well the ground truth and the negative action of Byzantine robots is stopped. Figure 3 shows that the absolute positional error is always very low in secured Swarm-SLAM, compared to the non-secured Swarm-SLAM case.

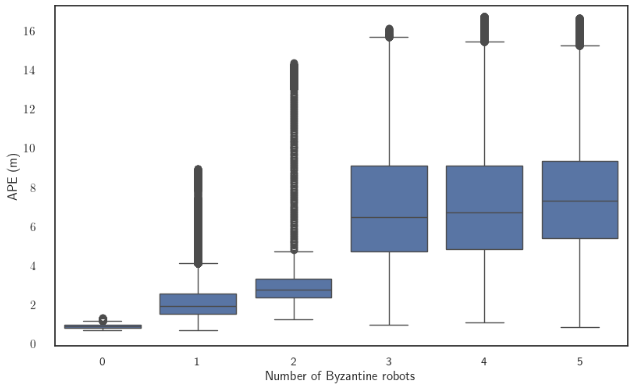

Figure 2: Performance in terms of Absolute Positional Error (APE) of a non-secured Swarm-SLAM system in case Byzantine robots injecting constant noise are presented. The overall APE is significantly bad. Moreover, the APE values increase noticeably with increasing number of Byzantine robots, which reaches values larger than 8 metres when just one Byzantine robot is present.

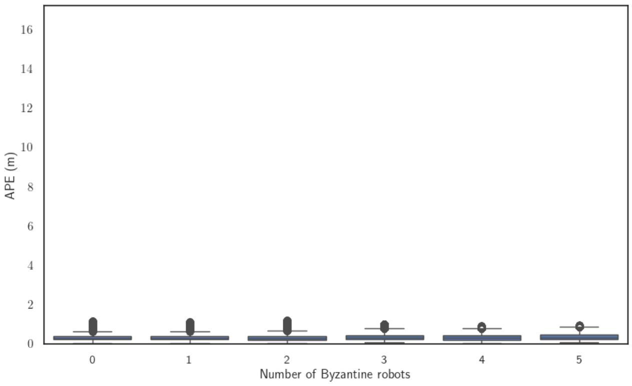

Figure 3: Secured Swarm-SLAM in case of constant noise perturbation. Notice how the APE is always very low. Moreover, the APE values never increase and the Byzantine behaviour is very well controlled. An almost constant noise is ever present in the map because of my simulations include noise in the odometry.

Further details about this type of plot and several additional results are discussed in the thesis.

5. Conclusion

The key findings of this thesis are:

1. An analysis of the robustness problem in SLAM systems with robot swarms when Byzantine robots participate to collective actions.
2. The development of a new Byzantine fault-tolerant framework for Swarm-SLAM secured by a blockchain-based smart contract.

In conclusion, this thesis has advanced our understanding of security vulnerabilities in SLAM systems with robot swarms by analysing the state-of-the-art framework, and proposed an effective solution to the problem of the creation of inter-robot loop closures by Byzantine robots. The findings not only highlight some problems not addressed yet in SLAM with robot swarms, but also have practical implications for secured Swarm-SLAM implementations. As we move forward in SLAM systems development, it is essential to study secure and robust architectures to continue advancing the field and addressing the complex challenges posed by robot swarms subjected to Byzantine robots.

6. Acknowledgements

I want to express my deepest gratitude to Dr. Andreagiovanni Reina, Dr. Volker Strobel and Ir. Alexandre Pacheco for providing me with the opportunity to delve into these topics and for guiding me during this research.

Thanks to Ir. Pierre-Yves Lajoie, PhD student from École Polytechnique de Montréal, for collaborating with me and for his constant availability.

A special thanks goes to Prof. Francesco Amigoni, that has been my internal supervisor at Politecnico di Milano.

I also express my gratitude to Prof. Marco Dorigo and all the people from IRIDIA for offering me the incredible opportunity to spend my days in their laboratory.

References

[1] Marco Dorigo, Guy Theraulaz, and Vito Trianni. Swarm Robotics: Past, Present, and Future [Point of View]. *Proceedings of the IEEE*, 109(7):1152–1165, 2021.

[2] Miquel Kegeleirs, Giorgio Grisetti, and Mauro Birattari. Swarm SLAM: Challenges and Perspectives. *Frontiers in Robotics and AI*, 8:1–16, 2021.

[3] Pierre-Yves Lajoie and Giovanni Beltrame. Swarm-SLAM : Sparse Decentralized Collaborative Simultaneous Localization and Mapping Framework for Multi-Robot Systems. *arXiv preprint*, 2023.

[4] Volker Strobel, Eduardo Castelló Ferrer, and Marco Dorigo. Blockchain Technology Secures Robot Swarms: A Comparison of Consensus Protocols and Their Resilience to Byzantine Robots. *Frontiers in Robotics and AI*, 7:54, 2020.

[5] Volker Strobel, Alexandre Pacheco, and Marco Dorigo. Robot swarms neutralize harmful Byzantine robots using a blockchain-based token economy. *Science Robotics*, 8(79):eabm4636, 2023.