sciendo

Int. J. Appl. Math. Comput. Sci., 2025, Vol. 35, No. 3, 519–533
DOI: 10.61822/amcs-2025-0037

amcs

MITIGATING CYBER–INTRUSIONS IN MEDICAL DEVICES WITH AGENT–BASED SELF–HEALING

ANA S. CARREON-RASCON [a,*], HUAYU LI [a], JERZY W. ROZENBLIT [a], WOJCIECH RAFAJŁOWICZ [b]

[a]Department of Electrical and Computer Engineering
University of Arizona
1230 E. Speedway Boulevard, Tucson, AZ 85721, USA
e-mail: {anascarreonr, jerzyr}@arizona.edu

[b]Faculty of Information and Communication Technology
Wrocław University of Science and Technology
Wyb. Wyspiańskiego 27, 50-370 Wrocław, Poland
e-mail: wojciech.rafajlowicz@pwr.edu.pl

Medical device security has become a critical focus in the healthcare sector, as increasing connectivity introduces challenges related to patient safety, data confidentiality, and system reliability. To address these concerns, various strategies have been developed, including risk identification, mitigation techniques, and autonomic recovery mechanisms. In this paper, we propose a novel conceptual framework that leverages reinforcement learning for self-healing in implanted medical devices (IMDs). This approach integrates automated recovery actions with real-time risk identification, providing a robust mechanism to maintain system functionality and safeguard patient well-being in the face of adversarial threats. By using a behavioral abstraction model of an insulin pump as a case study, our framework demonstrates the ability to maintain continuous system functionality under a variety of attack scenarios, achieving the maximum simulated survival time of 20,165 minutes for all cases. In comparison, without the self-healing mechanism, survival times drop significantly, particularly under attacks on critical components, such as glucose sensors and meters. These results highlight the effectiveness of the proposed approach in mitigating the impact of system failures and ensuring reliable operation of IMDs in adversarial environments.

Keywords: medical device security, reinforcement learning, self-healing, survival time.

1. Introduction

As technology continues to be implemented in various fields and solidifies its role as an indispensable part of modern life, its application in the medical sector has grown exponentially. This growth is exemplified by the advent of the Internet of Medical Things (IoMT). Similarly to the Internet of Things, the IoMT is characterized by incorporating internet connectivity. It is also characterized by its utilization of advanced technology for data analysis, processing and collection; swiftly revolutionizing healthcare by enhancing connectivity and enabling more efficient monitoring and management of patient care (Ahmed *et al.*, 2024; Deja *et al.*, 2021).

The integration of technology has significantly improved communication and data collection for both patients and medical practitioners. Wearable and implanted medical devices (IMDs) are prominent examples of this advancement. Equipped with sensors, these devices continuously monitor patients' health and provide critical data to users. In many cases, they also transmit this information directly to healthcare providers or hospitals, enabling timely interventions and improved care outcomes (Muhammad *et al.*, 2021).

Despite its benefits, the IoMT introduces substantial challenges, particularly in safety and security. Failures in life-critical devices, such as insulin pumps, pacemakers, and implantable cardioverter defibrillators (ICDs), pose

*Corresponding author

© 2025 A.S. Carreon-Rascon et al.
This is an open access article distributed under
the Creative Commons Attribution-NonCommercial-NoDerivs license
(http://creativecommons.org/licenses/by-nc-nd/4.0/).

serious threats to patient health. For instance, an insulin pump malfunction could lead to dangerous fluctuations in blood sugar levels, while the failure of a pacemaker or ICD during a cardiac event could result in life-threatening complications. Such risks highlight the pressing need for robust quality control measures and proactive device maintenance.

Beyond hardware malfunctions, IoMT devices are also exposed to cybersecurity threats. Vulnerabilities such as denial-of-service (DoS) attacks, data breaches, and injection attacks jeopardize the safety and privacy of patients, particularly when targeting devices that manage life-sustaining functions, such as IMD (Pritika *et al*., 2023). The potential for exploitation due to inadequate risk management or delayed security updates highlights the critical importance of implementing comprehensive security strategies (Baker, 2022).

To address these challenges, various automated fault-recovery strategies have emerged, including self-adaptivity, self-organization, reconfigurable systems, and self-healing (SH) approaches. Reconfigurable systems, primarily focusing on hardware-level fault recovery, may face limitations in tightly constrained IMDs where flexibility and redundancy are restricted (NRC, 2001). In contrast, SH systems integrate fault tolerance and self-stabilization with survivability, offering a dynamic solution. By following a detect-diagnose-recover loop, SH systems are well-suited for developing comprehensive schemes that address fault detection, mitigation, and recovery in IMDs (Psaier and Dustdar, 2011).

In recent years, the integration of Artificial Intelligence (AI) has revolutionized various domains, including healthcare and cybersecurity. Such examples are showcased by Fox *et al*. (2020) and Dénes-Fazakas *et al*. (2024), who leveraged reinforcement learning (RL) a tool to automatically calculate the amount of insulin to be given by a patient via an insulin pump. Furthermore, Wang *et al*. (2023) carried a proof of concept feasibility trial to evaluate a proposed RL approach towards glycemic control in type 2 diabetes patients. AI's capabilities in automating data processing, identifying patterns, and generating intelligent responses make it an ideal candidate for enhancing SH systems. By leveraging AI, particularly RL, SH systems can achieve advanced levels of automation and adaptability, addressing the inherent complexities of fault recovery and security management in IMDs.

In this work, we propose a novel approach that integrates SH with RL to enhance security and enable autonomous recovery in IMDs. To illustrate this approach, we focus on insulin pumps as a case study, demonstrating the feasibility and effectiveness as well as practical application of our methodology. This innovative scheme leverages the adaptability and learning capabilities of RL agents to autonomously detect, mitigate, and recover from potential faults or cyberattacks. By embedding intelligent decision-making into the SH framework, the system can respond dynamically to both expected and unforeseen disruptions, ensuring continuous device functionality and patient safety.

Insulin pumps are vital life-supporting devices that require robust mechanisms to maintain their reliability under strict performance constraints. Through this case example, we demonstrate how the RL-based SH approach can effectively manage risks, recover from disruptions, and uphold the integrity of critical health data in real time.

Additionally, we delve into the specific requirements for implementing agent-based SH in IMDs. This includes defining essential components such as fault-detection mechanisms, learning algorithms tailored to real-time healthcare scenarios, and resource-efficient recovery protocols that respect the physical and computational constraints of IMDs. Furthermore, we outline the principles for designing simulation environments that accurately model the complexities of IMD operations. These simulations are critical for testing and validating the effectiveness of the RL agent-based SH scheme, ensuring its readiness for deployment in real-world healthcare systems.

Through this comprehensive approach, we aim to establish a robust framework for integrating advanced AI-driven self-healing capabilities into life-critical medical technologies. The following sections are structured as follows: Section 2 will present related work with cyber-intrusions in IMDs, self-healing systems, and reinforcement learning; Section 3 introduces our methodology, exploring the requirements for self-healing, our RL formulations and risk mitigation scheme, and finally defining our agent-based Self-Healing approach and test-bench. In Section 4 we present our results, along with further information regarding our experimental setup and evaluation protocols. Finally, we present a discussion and conclusions obtained from this research.

2. Related works

2.1. Cybersecurity threats and vulnerabilities in implanted medical devices. As network connectivity becomes integral to the healthcare sector, the risks associated with cyberattacks have grown significantly. In 2024, global data breaches in healthcare cost an average of $4.88 million per incident, reflecting a 10% increase from 2023, according to a study by IBM (2024). However, implementing AI-driven security and automation measures has demonstrated promising outcomes, with average savings of $2.22 million per breach, highlighting the transformative potential of AI in bolstering security.

IMDs, a critical subset of the healthcare industry,

represent a growing market projected to reach $138.1 billion by 2030 (HGV Research, 2024a). This growth is driven by the increasing prevalence of chronic conditions such as diabetes and cardiovascular diseases, which necessitate continuous monitoring and treatment through devices like insulin pumps and pacemakers (HGV Research, 2024b). However, the incorporation of network connectivity into IMDs has amplified their vulnerability, raising significant concerns about patient safety and data security.

As previously mentioned, attacks on the healthcare sector have become widespread with the incorporation of network connectivity into the sector. However, $2.22 million savings on average were observed by the implementation of AI on security and automation, showcasing the positive impact of AI incorporation for security.

IMDs, by their very nature, are life-critical devices implanted within the patient's body to support vital functions. For example, implanted cardioverter-defibrillators (ICDs) regulate cardiac rhythms, while insulin pumps provide precise insulin delivery. The integration of connectivity in these devices exposes them to risks ranging from unexpected failures to data leaks and cyberattacks (Hassija *et al*., 2021). These vulnerabilities have prompted regulatory bodies such as the US Food & Drug Administration (FDA) to mandate security measures for IMDs FDA (2025; 2023), and innovative approaches like authentication techniques combined with proximity sensing (e.g., distance bounding protocols) have been proposed to mitigate risks, such as man-in-the-middle attacks (Camara *et al*., 2021). Some methods even derive security keys from physiological signals to enhance protection (Pirbhulal *et al*., 2018).

The unique risks associated with IMDs require comprehensive threat modeling and risk assessment strategies, such as STRIDE (Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of Privilege). High-risk IMDs, particularly class III devices like pacemakers and ICDs, face distinct challenges. For instance, battery depletion attacks have been a long-standing concern, as highlighted in 2017 by the voluntary recall of nearly 500,000 implantable pacemakers due to cybersecurity vulnerabilities (Kuehn, 2018).

Data confidentiality and integrity are additional critical vulnerabilities. IMDs often store and transmit sensitive patient information to healthcare providers. Without robust authentication mechanisms and access controls, this information becomes susceptible to breaches. Unauthorized access could compromise not only data confidentiality, but also data integrity, enabling potential tampering with device functionality.

Network vulnerabilities further exacerbate these risks, as insecure communication channels can be exploited to disrupt device operations. However, while enhancing security is imperative, it must be balanced with the need for medical practitioners to access IMDs easily during emergencies. This showcases the complexity of securing IMDs and highlights the need to navigate trade-offs between robust security and operational accessibility.

In conclusion, cyber intrusions in IMDs pose multifaceted challenges that demand innovative, multi-layered solutions. Addressing these threats requires a combination of advanced security measures, regulatory compliance, and thoughtful design principles that prioritize both patient safety and device functionality.

2.2. Advancing system resilience with self-healing. SH systems are designed to detect, diagnose, and recover from faults autonomously. Recovery in SH systems does not necessarily imply a complete restoration of functionality; rather, it may involve some loss in performance, as noted by the term *degradation* introduced by Koopman (2003). This concept refers to a system's ability to maintain partial functionality by prioritizing critical tasks over non-vital ones. Common recovery methods include disabling less important tasks, rebooting from a previous checkpoint, and other similar approaches.

A notable example of an integrated SH system is described by Seiger *et al.* (2015; 2018), who present PROtEUS, a process execution system that utilizes closed feedback loops, specifically the MAPE-K (Monitor, Analyze, Plan, Execute, and Knowledge) framework. This system highlights the importance of addressing inconsistencies between software evaluations and real-world outcomes, emphasizing the necessity of accurate detection and correction of these discrepancies for effective SH implementation.

In another study (Dong *et al*., 2003), an SH computing environment is introduced, built upon a process comprising *monitoring, analysis and verification, and adaptation*. This system employs a fault handler tailored for different fault types, such as system-level, component-level, or agent-level faults. During the monitoring phase, the fault handler detects anomalies and triggers the subsequent analysis and verification phase. Here, a SH handler identifies the nature of the fault and its recovery requirements. In the adaptation phase, the fault handler executes the appropriate recovery procedures or consults an application-delegated manager to employ alternative resources, such as switching to a backup machine, to ensure continuity.

AI further enhances the capabilities of SH systems, particularly in anomaly detection and fault prediction. By training models on historical data to establish baseline system behavior, AI systems can identify deviations from normal patterns, flagging potential issues before they escalate. This predictive functionality not only

helps in early fault detection, but also assists in root cause analysis, enabling proactive measures to prevent recurrence (Johnphill *et al*., 2023).

In summary, SH systems represent a critical advancement in fault tolerance, with applications ranging from process management to embedded systems. By integrating AI, these systems can achieve heightened levels of automation, accuracy, and efficiency, making them a cornerstone of resilient computing environments.

2.3. Reinforcement learning. RL has emerged as a powerful tool for addressing complex decision-making problems, particularly in environments marked by uncertainty, dynamic interactions, and long-term objectives. Unlike supervised learning, which relies on labeled data to train models, RL empowers an agent to learn autonomously by interacting with its environment. Through this process, the agent receives feedback in the form of rewards or penalties, enabling it to optimize its behavior over time to achieve a desired outcome (Sutton and Barto, 2018). This trial-and-error learning mechanism allows RL to uncover strategies that are not explicitly programmed, making it uniquely suited for tasks with intricate dependencies and sequential decision-making requirements. RL is particularly suitable for environments where the optimal strategy cannot be predefined and must be discovered through exploration and exploitation of possibilities.

RL has been successfully implemented across a variety of domains requiring intelligent decision-making capabilities. As an example, it has been utilized to facilitate real-time decision-making for navigation, collision avoidance, and route optimization in self-driving cars (Sallab *et al*., 2017). These successes highlight RL's capacity to operate effectively in scenarios where conventional methods fall short, such as those involving delayed rewards, high-dimensional state spaces, or evolving environments.

This versatility of RL extends to healthcare, where it has been explored for applications such as treatment planning, medical imaging analysis, and personalized medicine (Yu *et al*., 2018). RL enables data-driven decision-making that allows for patient-specific needs, improving the precision and effectiveness of medical interventions. In the context of IMDs, RL has potential to enhance their functionality, security, and resilience by

- Fault Recovery: RL can optimize self-healing strategies by learning efficient recovery actions tailored to device-specific constraints.
- Resource Efficiency: IMDs face tight constraints when it comes to computational and power limitations. RL can intelligently allocate resources, ensuring minimal power consumption while maintaining operational accuracy.
- Security Enhancements: For instance, RL has been proposed to improve the reliability of insulin pumps by classifying insulin dosages as genuine or false, reducing risks of misadministration and enhancing patient safety (Rathore *et al*., 2020).

In this way, the adaptability of RL makes it particularly suitable for IMDs, enabling continuous learning and improvement over time. As patient conditions or device usage patterns evolve, RL allows IMDs to dynamically respond to these changes, maintaining optimal performance and mitigating potential risks. Its ability to automate complex decision-making processes while accounting for operational constraints makes RL a key technology for advancing the reliability and security of medical devices. In conclusion, RL represents a cutting-edge approach to addressing challenges in IMDs and beyond. Its ability to navigate uncertainty, adapt to evolving conditions, and optimize long-term outcomes positions RL as a cornerstone technology for the next generation of intelligent and resilient systems.

It happens that RL approaches are often a source of complaints by system designers as they require extensive computations and a large amount of memory. Fortunately, in IMDs these requirements can be largely reduced by the following factors:

- The number of IMDs states is finite and not very large.
- The number of automated recovery actions is also finite and frequently less than the number of states.
- If the number external or internal risks states of an IMD is finite, then we are able to pre-compute good policies for this type of IMD on a main-frame computer and to transfer the results of learning to the IMD. Later, these policies are individualized to a given patient by further RL.
- When the number of the risks states is infinite or very large, one can approximate the policy function for a selected type of the IMD on a main-frame computer and transfer it, largely reducing memory requirements.

The reader may find more on these topics in the works of Chen *et al*. (2019) and Zabihi *et al*. (2023), where similar aspects arising in the internet of things are discussed, as well as in that of Kegyes *et al*. (2021), where computational aspects of RL applied in the industry 4.0 can be found. A general survey is presented by Rafajłowicz (2022).

3. Methodology

3.1. Requirements for agent-based self-healing of implanted medical devices. Run-time threat detection

and risk probability estimation are crucial for enhancing the security and reliability of IMDs. Previous research has introduced non-intrusive methods for threat detection and risk probability estimation, leveraging timing samples collected from a device's trace port (Rao *et al.*, 2017). These techniques have effectively supported multi-modal software design and adaptive risk modeling, enabling automatic threat mitigation. However, they fail to address recovery actions, which are an essential component of comprehensive SH systems.

Carreon-Rascon and Rozenblit (2022) proposed a set of requirements for SH systems was, combining authentication and mitigation schemes to enable automated recovery actions in IMDs. The integration of RL agents into SH systems for IMDs introduces unique opportunities for adaptability and intelligence, but it also brings additional challenges. To address these, agent-based SH systems must satisfy several domain-specific requirements:

- **Least Intrusive Monitoring**: Monitoring mechanisms in IMDs must avoid interfering with device functionality or compromising patient safety. Conventional approaches often involve direct access to system instructions or memory, which can disrupt operations. Instead, non-intrusive techniques, such as utilizing trace port data, ensure effective oversight without compromising performance.
- **Least Harmful Recovery Actions**: Recovery mechanisms must prioritize safety and predictability. Rather than allowing RL agents to directly modify system instructions, recovery actions should utilize predefined reset interfaces. These interfaces restore the device to a validated safe state, reducing the risk of unintended consequences and maintaining system integrity.
- **Precise Component and Functionality Identification**: Efficient and targeted recovery is essential. Thus, classification algorithms play a critical role in pinpointing the impacted areas, ensuring recovery efforts are directed where they are most needed.
- **Risk-Based Recovery Strategies**: Recovery actions must be adaptable and proportional to the severity of the threat or fault. For minor faults, localized reconfigurations may suffice, while more critical disruptions may necessitate reverting to a previously validated safe state. This risk-based approach optimizes recovery efforts and resource utilization.

By combining the foundational principles of traditional SH systems with the advanced capabilities of RL agents, agent-based SH systems for IMDs can address the unique demands of these life-critical devices. This hybrid approach ensures robust fault detection, precise diagnostics, and controlled recovery, all while respecting the safety requirements and resource constraints inherent to IMDs. Through this enhanced framework, SH systems can become a cornerstone of resilient and secure medical device design, safeguarding both patient health and operational reliability.

3.2. Principles of simulation design. In this research, we utilize a behavioral abstraction to model an IMD and its associated environment. The model is designed to include all key components and functionalities of an IMD, ensuring a comprehensive representation of its operational environment. Each primary component of the IMD (e.g., sensors, human interface, controller) is instantiated with predefined characteristics and behaviors. Additionally, the simulation incorporates representations of the patient, to generate physiological signals and simulate responses to the IMD's actions, and an adversary, to simulate different types of cyber attacks.

We propose a general simulation framework that includes four main components, each serving a distinct role:

- **Interactive Environment (IMD)**: The IMD itself acts as the interactive environment, comprising various interconnected components such as sensors to monitor physiological data, a controller for decision-making and actuation, and interfaces for human interaction. These elements collectively mimic the IMD's functionality in real-world scenarios, capturing both normal operations and responses to threats.
- **Passive Environment (Patient)**: The patient is modeled as a passive entity generating physiological signals, such as glucose levels for an insulin pump or cardiac rhythms for a pacemaker. The patient component interacts with the IMD, providing input data for sensors and receiving outputs, such as insulin injections from the IMD.
- **Adversary**: The adversary represents a range of potential cyber threats, simulating various types of attacks that the IMD may encounter. These include denial-of-service (DoS) attacks, data tampering, injection attacks, or battery depletion threats. By modeling diverse attack scenarios, the adversary component challenges the IMD's robustness and tests the SH system's effectiveness.
- **Agent (Self-Healing Mechanism)**: The agent embodies the SH mechanism, designed to detect, diagnose, and recover from faults or attacks autonomously. It interacts with the IMD, analyzing trace port data, identifying anomalies, and implementing recovery actions based on predefined strategies or learned behaviors.

To ensure that the simulation effectively models real-world conditions while facilitating robust testing of the proposed SH framework, several core principles are followed:

- **Modularity**: The simulation is structured into modular components, allowing for flexibility in updating or replacing specific elements. For example, different patient models can be integrated to simulate various physiological conditions, or adversary models can be modified to represent emerging attack vectors.
- **Realism**: Each component is designed to mimic real-world behaviours as closely as possible. Physiological signals generated by the patient model are based on empirical data, and attack scenarios are informed by documented vulnerabilities and threats in IMDs. This ensures that the simulation provides meaningful insights into the system's performance in practical applications.
- **Scalability**: The simulation is built to accommodate varying levels of complexity, from simple IMD operations to scenarios involving multiple adversaries or complex patient conditions. This scalability allows for progressive testing and evaluation of the self-healing mechanism under diverse conditions.
- **Interactivity**: The dynamic interaction between components (e.g., IMD, patient, adversary, and agent) is a central feature of the simulation. The agent's ability to detect and respond to threats depends on real-time data exchanges and feedback loops, ensuring a realistic representation of operational challenges.
- **Adaptability**: The framework supports integration with machine learning models, such as reinforcement learning agents, to enhance self-healing capabilities. This adaptability ensures the simulation remains relevant as new algorithms and techniques are developed.

By adhering to these principles and incorporating additional features, the simulation provides a comprehensive platform for designing, testing, and refining agent-based SH systems for IMDs.

3.3. Reinforcement learning agent-based self-healing for implanted medical devices. The general workflow for the RL Agent-based SH for IMDs is showcased in Figure 1. It includes two core components: a risk assessment network f_{risk} and a control network f_{control}. Together, these components work to estimate the system's risk and determine optimal recovery actions to ensure the reliable, secure, and continuous operation of the IMD. The overall training process is divided into two sequential stages, first a supervised learning for risk estimation and secondly RL for control optimization.

The risk assessment network leverages a transformer architecture (Vaswani *et al.*, 2017) to estimate the probability that the system is under attack or in a compromised state. Leveraging from this architecture (showcased in Fig. 1) allows us to model temporal correlations in our data. This network uses the state of the IMD, represented as $S_{t-T:t}$, which is a time-series vector capturing operational data from various components over a sliding time window of size T. The input state $S_{t-T:t}$ includes operational data from all M components, defined as

$$S_{t-T:t} = \left\{ s^i_{t-T:t} : t \right\}_{i=0}^{M}, \tag{1}$$

where $s^i_{t-T:t}$ represents the data from component i during the time interval $[t-T, t]$. We denote $\mathcal{S}$ as the state space, which is the set of all possible state sequences $S_{t-T:t}$.

The network outputs a risk probability, $r_t \in [0, 1]$, representing the likelihood of the system being in a vulnerable state:

$$r_t = f_{\text{risk}}(S_{t-T:t}; \theta), \tag{2}$$

where θ are the learnable parameters of the network. This output serves as a quantitative measure of system vulnerability.

To train this network, supervised learning is employed with labeled data to minimize the binary cross-entropy loss:

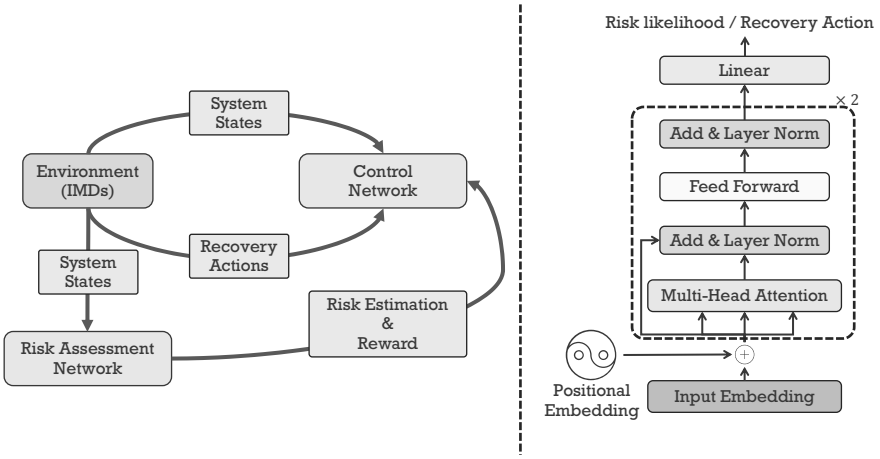
$$L_{\text{risk}} = -\frac{1}{N} \sum_{n=1}^{N} [y_n \log r_t + (1 - y_n) \log(1 - r_t)], \tag{3}$$

where y_n is the ground-truth label indicating whether the system is under attack, and N is the total number of training samples. This process ensures the network accurately estimates the system's current risk, which is crucial for informing the control network.

The control network (i.e., our RL agent) determines the optimal recovery actions to mitigate risks and maintain system functionality. Similarly to the risk assessment network, the control network leverages a transformer architecture to process the time-series state $S_{t-T:t}$, but focuses instead on generating an action vector denoted by A_t. This vector comprises binary decisions for each IMD component, indicating whether a corrective action (e.g., a reset) should be applied:

$$A_t = \left\{ a^i_t \right\}_{i=0}^{M}, a^i_t \in \{0, 1\}. \tag{4}$$

A value of $a^i_t = 1$ signifies that a recovery action is applied to component i, while $a^i_t = 0$ indicates no action



(a) (b)

Fig. 1. Self-healing system with the reinforcement learning scheme: the overall framework of the self-healing system for implanted medical devices (IMDs) consists of a risk assessment and a control network. The risk assessment network evaluates the system's states and estimates current risk, while the control network determines recovery actions based on feedback. These components operate in a loop to maintain system stability by continuously monitoring and mitigating risks (a); the architecture of the transformer-based model used in the risk assessment and control networks. The model includes input embeddings, positional encoding, multi-head attention mechanisms, feed-forward layers, and normalization to estimate risks and determine corrective actions effectively (b).

is taken. Let $\mathcal{A}$ denotes the action space, which is the set of all possible action A_t.

The control network is trained using reinforcement learning to maximize a reward function designed to encourage effective risk reduction while penalizing unnecessary corrective actions. The reward rew_t at time t is defined based on the current risk r_t and and previous risk r_{t-1} (outputs of risk assessment network f_{risk}) as

$$\text{rew}_t = \begin{cases} A & \text{if } |r_t - r_{t-1}| \leq \lambda, \\ B & \text{if } |r_t - r_{t-1}| > \lambda \quad \text{and} \quad r_t < r_{t-1}, \\ C & \text{if } |r_t - r_{t-1}| > \lambda \quad \text{and} \quad r_t > r_{t-1}, \end{cases} \tag{5}$$

where A, B, and C are constants representing rewards for stable, reduced, and increased risk, respectively (i.e., $B > A > C$), and λ is a threshold for significant risk changes.

The control network learns a stochastic policy as defined by Kala (2024), denoted π_ϕ, which is parameterized by ϕ. The policy π_ϕ, is a function mapping from the state space $\mathcal{S}$ to the space of probability distributions over the action space $\mathcal{A}$, denoted $P(\mathcal{A})$. Formally,

$$\pi_\phi : \mathcal{S} \to P(\mathcal{A}). \tag{6}$$

For any given state $S_{t-T:t} \in \mathcal{S}$, where T represents the time window; $\pi_\phi(S_{t-T:t})$ yields a probability distribution over all possible actions $A' \in \mathcal{A}$. The probability of selecting a specific action $A_t \in \mathbf{A}$ in state $S_{t-T:t} \in \mathcal{S}$ under policy π_ϕ is denoted by $\pi_\phi(A_t|S_{t-T:t})$. This is a scalar value representing this specific probability. The list of actions is centered around resetting the components to work under a set of given instructions that ensure each component is performing the necessary tasks (e.g., sensor storing and passing on data to the controller, controller calculating insulin dosage, insulin dosage transmitted to the insulin pump, etc.).

The objective of the reinforcement learning agent is to find the policy parameters ϕ that maximize the expected cumulative discounted reward. This objective function, $J(\phi)$, is defined as

$$J(\phi) = \mathbb{E}_{\tau \in \pi_\phi}[\sum_{k=t_0}^{H} \gamma^{k-t_0} \text{rew}_k]. \tag{7}$$

Here, τ represents a trajectory of states, actions, and rewards, i.e.,

$$\begin{aligned} &S_{t_0-T:t_0}, A_{t_0}, \text{rew}_{t_0}, \\ &S_{(t_0+1)-T:t_0}, A_{(t_0+1)}, \text{rew}_{t_0+1}, \\ &\vdots \\ &S_{H-T:H}, A_H, \text{rew}_H. \end{aligned}$$

The trajectory is generated by the agent following policy π_ϕ within the environment. The expectation $\mathbb{E}_{\tau \in \pi_\phi}$ is taken over all possible such trajectories. rew_k is the scalar reward obtained at timestep k as defined in Eqn. 5, $\gamma \in (0, 1]$ is the discount factor, t_0 is the initial timestep of an episode, and H is the episode horizon.

To optimize $J(\phi)$ using a gradient ascent method (as part of the policy gradient method), we need its gradient

with respect to ϕ. According to the Policy Gradient Theorem (Kala, 2024), this gradient $\nabla_\phi J(\phi)$ is given by

$$\nabla_\phi J(\phi) = \mathbb{E}_{\tau\sim\pi_\phi}\left[\sum_{t=t_0}^{H}\left(\nabla_\phi \log \pi_\phi(A_t|S_{t-T:t})\right) G_t\right]. \tag{8}$$

In this expression,

- $S_{t-T:t}$ is the state sequence observed at timestep t;
- A_t is the action taken at timestep t, sampled according to the policy's probability distribution $\pi_\phi(\cdot|S_{t-T:t})$;
- $\pi_\phi(A_t|S_{t-T:t})$ is the specific probability of selecting action A_t in state $S_{t-T:t}$ under policy π_ϕ;
- $\nabla_\phi \log \pi_\phi(A_t|S_{t-T:t})$ is the gradient of the natural logarithm of this action probability with respect to the policy parameters ϕ. This term indicates how to adjust ϕ to increase the log-probability of action A_t;
- G_t is the discounted cumulative future reward (also known as the return) starting from timestep t:

$$G_t = \sum_{k=t}^{H} \gamma^{k-t} \text{rew}_k. \tag{9}$$

  This sum G_t weighs future rewards according to the discount factor γ, accounting for the fact that rewards obtained sooner are generally more valuable. The gradient term $\nabla_\phi \log \pi_\phi(A_t|S_{t-T:t})$ is then weighted by this return G_t.

The expectation $\mathbb{E}_{\tau\sim\pi_\phi}$ signifies that, in practice, this gradient is estimated by averaging the quantity $\sum_{t=t_0}^{H}\left(\nabla_\phi \log \pi_\phi(A_t|S_{t-T:t})\right) G_t$ over multiple trajectories sampled by executing the policy π_ϕ.

The combination of supervised learning for risk estimation and RL for control optimization ensures a robust SH mechanism for IMDs. The risk assessment network provides timely and accurate vulnerability metrics, while the control network mitigates threats through adaptive recovery strategies. This enhances the reliability, security, and operational resilience of IMDs, which is crucial for maintaining patient safety in dynamic and potentially adversarial environments.

3.4. Overview of the proposed architecture. Our system's architecture is designed to compute threat probabilities from sequential trace port data. For this purpose, we utilize a transformer-based neural network to leverage their self-attention ability to learn context, capture complex patterns, and model temporal dependencies in sequential data. Unlike methods such as sliding windows that may truncate important contextual information, the transformer's encoder can process and retain relationships across extended data sequences. As shown in Fig. 1, our approach utilizes only the encoder layers of the standard transformer architecture (Vaswani *et al.*, 2017); we omit the decoder layers as our goal is to analyze and classify the input sequence rather than generate a new one.

A key component of the transformer architecture is its self-attention mechanism, which allows the model to weigh the importance of different parts of the input sequence when processing a specific position. The input for an attention layer consists of three matrices from the embedding of the trace port data: Queries (Q), Keys (K), and Values (V). In the encoder, these are all derived from the output of the previous layer. The attention score is calculated using the following scaled dot-product formula:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V, \tag{10}$$

where d_k is the dimension of the keys, and the scaling factor $1/\sqrt{d_k}$ prevents the dot products from becoming too large.

This attention mechanism is further enhanced by employing Multi-Head Attention, which involves running the attention mechanism multiple times in parallel with different learned linear projections of the original Q, K, and V matrices. With this, the model attends to information from different representational subspaces at different positions in parallel. The outputs are then concatenated and once again projected, resulting in the final values. The process is defined as

$$\begin{aligned}&\text{MultiHead}(Q, K, V)\\ &\qquad = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^o,\end{aligned} \tag{11}$$

where

$$\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^K) \tag{12}$$

and $W_i^Q, W_i^K, VW_i^K, W^o$ are learnable parameter matrices of linear projections.

Our model is composed 2 encoder layers, both composed of two primary sub-layers: the multi-head self-attention mechanism described above, followed by a position-wise, fully connected feed-forward network, in this case a Multi-Layer Perceptron (MLP). Layer normalization is applied around each of the two sub-layers to facilitate effective training. For our specific implementation, the transformer model uses 6 attention heads in the multi-head attention mechanism, and a hidden dimension of 96.

3.5. Test bench: A simulated insulin pump. To evaluate the proposed SH framework, a behavioral abstraction of an insulin pump was developed as a case study, as depicted in Fig. 2. This abstraction models the essential components and functionalities of an insulin pump system, ensuring a close representation of real-world operations. The main components and their roles are shown below:

- **Glucose sensor**: A component of the Continuous Glucose Monitoring (CGM) system. It is responsible for measuring glucose levels in the interstitial fluid surrounding the patient's cells. This near real-time data serves as the foundation for insulin delivery decisions. Functionality includes measuring glucose levels with optional Gaussian noise to simulate variability, and performing auxiliary tasks such as read/write operations, and data integrity checks.

- **Blood glucose meter**: this component simulates the analog to digital glucose level, which transform the sensor reading to blood glucose levels.

- **Data hub & human interface**: Acts as an intermediary between the sensors and the controller. It processes and transmits sensor data while allowing user input. Functions include reading, writing, and maintaining the synchronization of data streams, ensuring seamless communication within the system.

- **Controller**: Processes sensor data and calculates the required insulin infusion rate based on current glucose levels, insulin sensitivity, and programmed parameters. It ensures precise control to maintain the patient's glucose levels within healthy ranges.

- **Pump**: Implements the insulin infusion by executing the instructions from the controller. The pump dynamically adjusts its status (e.g., "Idle" or "Delivering") based on insulin requirements. It performs self-checks and mutex locks to ensure safe operation.

- **Infusion set**: Directly delivers insulin to the patient's body. The infusion set verifies insulin delivery operations and maintains logs for traceability. Safety measures include locking mechanisms to prevent accidental over-delivery.

Each component has predefined instructions specifying its default operational mode. These instructions can be reset via an interface controlled by the RL agent, enabling a return to the default state when necessary.

The test bench also incorporates a simulation model for the patient in order to emulate human physiological responses, enabling comprehensive validation of the system. The simulation accounts for:

- **Meal Intake Simulation**: Models the patient's eating patterns, incorporating meal schedules, carbohydrate content, and glucose uptake rates. The system generates glucose input dynamically during meal times.

- **Blood Glucose Dynamics**: Simulates fluctuations in glucose levels based on food intake and insulin infusion. The model employs differential equations to capture the interaction between glucose and insulin levels, adapting to both meal events and basal conditions.

- **CGM Feedback Loop**: Produces real-time glucose data in order to emulate the CGM system; produces measurements at five-minute intervals. This feedback enables timely insulin adjustments to maintain glucose levels within the target range.

To test the robustness of the SH framework, the test bench incorporates an adversary simulation module to introduce faults or tamper with system components. Example attack scenarios include:

- **Sensor Tampering**: Disabling the glucose sensors to prevent measurements, and thus rendering it incapable of detecting glucose levels.

- **Meter Tampering**: Compromising the blood glucose meter, thus disrupting accurate glucose level measurements.

- **HID Attacks**: Disabling data transmission between components, causing systemic communication failures.

- **Controller and Pump Disruptions**: Preventing insulin rate calculations or delivery operations, potentially leading to hypo- or hyperglycemia.

To expand upon this, each component works by following a set of steps or instructions that need to be followed to perform their given tasks. For example, a sensor must read and write the glucose measurement. If this instruction is tampered with, other components are not able to access the stored data, disrupting the controller's calculation thread from calculating the correct insulin dose.

For the set of corrective actions we have defined a set of predefined instructions for each component. This allows our corrective actions to reestablish the component's functionalities.

Finally, the adversary records the attack history and dynamically determines the feasibility of subsequent attacks, enabling realistic tampering scenarios for our evaluation.

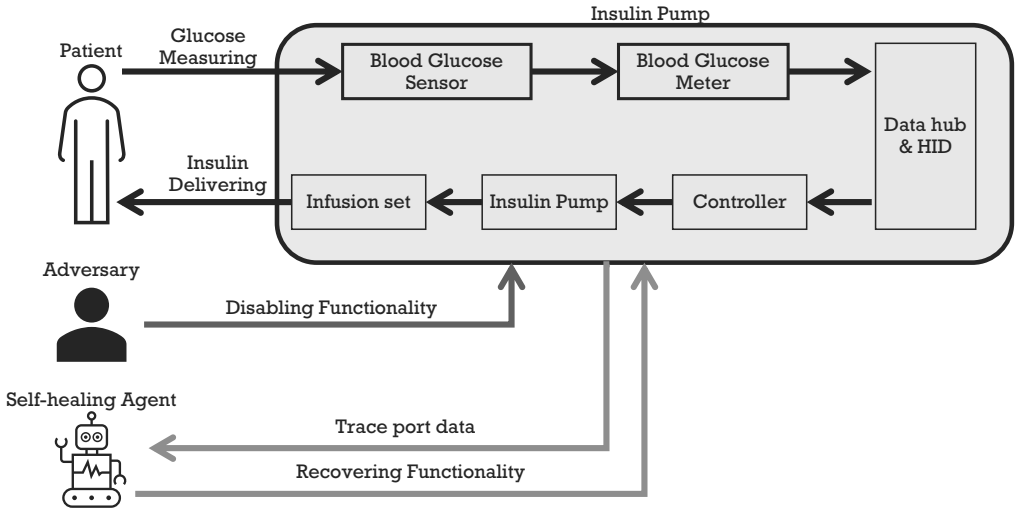


Fig. 2. Insulin pump behavioral abstraction. The figure illustrates the operational components of an insulin pump system, including the blood glucose sensor, blood glucose meter, controller, insulin pump, infusion set, and data hub & HID. The adversary disrupts the system by disabling functionalities (medium-grey arrow), while the self-healing agent counteracts these disruptions by recovering functionalities (light-grey arrows).

4. Results

4.1. Experimental setup. The experimental setup involves two aspects, the configuration and the evaluation of the risk assessment network, as well as the control network; each of these networks is designed to address specific aspects of the self-healing framework. For the risk assessment network, two datasets were generated to model normal and abnormal behaviours. More specifically, normal data represents typical system behavior without interference, while abnormal data captured scenarios where attacks disrupted operations. Abnormal data collection spanned the time from the start of each simulation run to the point of system failure (e.g., hyperglycemia due to sensor deactivation). Each two-week simulation run yielded 100 samples, creating a comprehensive dataset to represent both types of behaviour. This dataset was then divided into a training set (80%) and a validation set (20%). Training was performed using the Adam optimizer (Kingma, 2014) with a learning rate of 0.001, a batch size of 256, and a binary cross-entropy loss criterion. The network was trained for 10 epochs to ensure convergence.

The control network was evaluated without splitting the data into training and validation subsets, as it relied on continuous trace port data. This network used the pre-trained risk assessment network as a critic to optimize corrective actions. Similarly to the risk assessment network, training for the control network was conducted with the Adam optimizer at a learning rate of 0.001 over a simulated period of one week (7 days $\times$ 1440 minutes/day). Trace port data was sampled 12 times per hour (to represent the sampling of the CGM), providing fine-grained inputs to the control network. The training involved 1000 episodes and used a discount factor γ of 0.99 to prioritize long-term rewards. The control network architecture consisted of two layers designed to process feedback from the risk assessment network effectively.

The transformer model used in both networks is comprised of two layers with positional encoding to handle sequential data. It incorporated six attention heads for efficient processing of input features and utilized a dropout rate of 0.1 to prevent over fitting. This architecture ensured robust temporal and contextual understanding of the trace port data.

4.2. Risk assessment network training. To determine an appropriate number of epochs for training, we conducted a training session of 20 epochs and monitored the model's performance on a validation set by observing the F1-score values obtained. As illustrated in Fig. 3, the model showed rapid performance improvements in the initial epochs, reaching an F1-score above 0.9 by epoch 2 and reached close to its maximum by epoch 5. Beyond epoch 10, we observed only marginal gains, and performance remained mostly stable.

Based on these observations, we set training to 10 epochs for the final model. This choice was made to balance computational efficiency with model performance, avoiding overfitting while ensuring convergence. As the network's purpose is to provide timely risk assessments during RL training, early convergence and stability were prioritized.

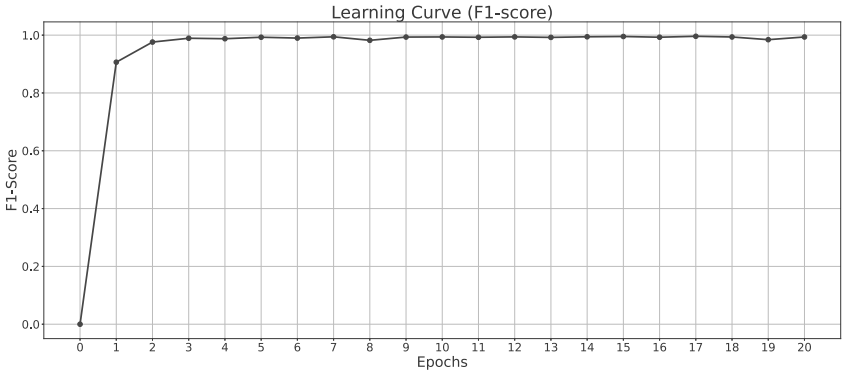


Fig. 3. Learning curve showing F1-score of the risk assessment network over 20 training epochs. Performance plateaus after epoch 10, justifying early stopping.

4.3. Evaluation protocols. Evaluating the safety and resilience of an IMD is of upmost importance, specially its ability to maintain life-critical functions under adverse conditions. Ensuring the IMD's capacity to deliver these life critical services, even in the presence of continuous threats, is a critical measure of its reliability. Inspired by risk mitigation strategies such as mode switching to isolate non-vital functionalities, the safest operational approach during sustained threats involves defaulting to predefined values established by medical practitioners. These default settings aim to ensure the continued delivery of life-sustaining functions to the patient.

For the introduction of attacks during the evaluation phase, To assess the performance of our agent-based SH framework, we conducted simulations that spanned a two week period while introducing a specific type of attack during this time.

To be more specific, an attack is generated for each simulation at start time. For each two week period, each five minute time step a new attack is generated based on random probability given that the previous threat was successfully recovered from. With this approach, we aim at simulating an environment under continuous threats to test the resiliency and reaction of our system.

For each attack scenario, a series of fifty test iterations were carried out. The average survival time of the patient, meaning the run-time for the simulation to continue without a health related interruption ending the simulation early, for each repetition is recorded to evaluate the effectiveness of the agent's actions. As a baseline, the same procedure is performed without employing the SH framework, allowing for a direct comparison.

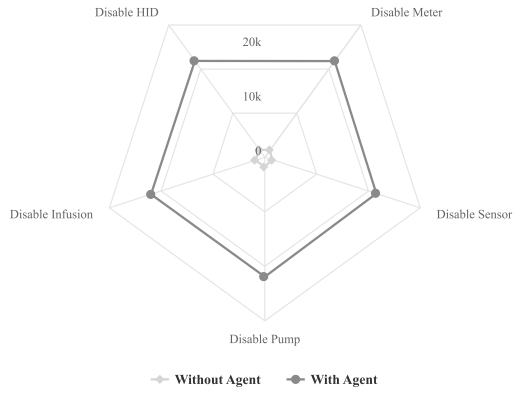
This evaluation method reflects our hypothesis that disabling critical functionalities without a mechanism for recovery will adversely affect the patient's health, leading to shorter survival times. In contrast, the integration of our SH agent is expected to mitigate the impacts of attacks, thereby ensuring patient survivability, demonstrating the robustness of the proposed framework.

4.4. System stability with the RL system. The evaluation results are presented in Table 1, they provide a comprehensive overview of the system's performance under different attack scenarios. From this result, we can make two key observations. First, the results demonstrate the effectiveness of the RL agent in ensuring system stability and patient survival. Across all simulated attack scenarios, the RL agent successfully mitigates the impacts of the attacks, enabling the simulation to reach its maximum runtime of 20,165 minutes without failure. This highlights the RL system's capability to accurately detect the attack patterns and implement corrective measures that sustain the IMD's critical functionalities. In contrast, the survival times without the agent (shown in red on the radar chart) showcase a significant decrease, reflecting the consequences of unmitigated attacks on the patient's health. This disparity proves the RL agent's role in safeguarding the IMD.

Secondly, the results allow us to categorize the criticality of each attack based on the survival times recorded in the absence of the agent. Among the various attack types, disabling the glucose meter has the most severe impact, leading to the shortest survival times. This indicates the crucial role of blood glucose sensing in maintaining precise insulin delivery. Faulty or absent glucose readings can cause miscalculated insulin doses, resulting in life-threatening hyper- or hypoglycemia. Conversely, tampering with the infusion pump has the least impact on survival time. This suggests that delays in insulin delivery, while suboptimal, are less immediately detrimental compared to inaccurate glucose measurements. Delays may allow some insulin regulation to continue, although at a reduced efficacy, minimizing the immediate risk to the patient.

Table 1. Average survival times observed in the study under two conditions: with and without the reinforcement learning-based self-healing framework. The results highlight the system's performance and resilience across various attack scenarios, and are visually showcased in radar chart (below).

Attack	With agent	Without agent
Disable Sensor	20165.00 ± 0.00	1652.40 ± 943.20
Disable Meter	20165.00 ± 0.00	1357.20 ± 489.60
Disable HID	20165.00 ± 0.00	1753.20 ± 1303.76
Disable Infusion	20165.00 ± 0.00	1796.40 ± 913.04
Disable Pump	20165.00 ± 0.00	1688.40 ± 1082.78



These findings showcase the nuanced interactions between the IMD's components and their role in sustaining patient health. Sensor tampering has a direct and critical effect, leading to rapid health deterioration, whereas actuator tampering, such as with the infusion pump, introduces delays that are less harmful over the short term.

These results highlight the need for continued investigation into these dynamics. For example, understanding why sensor tampering results in significantly shorter survival times could guide the prioritization of defense mechanisms. Similarly, exploring whether specific mitigation strategies could reduce the effects of actuator tampering could further improve the system's resilience.

In conclusion, the RL-based SH framework demonstrates robust performance, achieving maximum simulated survival time under all attack scenarios. The system effectively stabilizes IMD operations, even in the face of critical attacks. These findings help pave the way for future research aimed at optimizing IMD defenses and deepening our understanding of attack criticality across different system components.

5. Discussion

In this study, we introduce a conceptual idea and its early implementation for a SH approach for IMD, utilizing RL. This approach aims at addressing automatic recovery for IMD, leveraging from previous studies to incorporate an all encompassing approach (identification, mitigation, and recovery) to safeguard these devices without the need for direct human intervention.

The results of this study highlight the potential of RL as a robust framework for SH in IMDs. By evaluating survival times under various attack scenarios, we demonstrated that the RL-based system consistently maintained system stability and patient safety, achieving maximum simulated survival time in all tested cases. These findings highlight the framework's ability to identify, mitigate, and recover from system disruptions effectively, even under adversarial conditions.

The analysis of the attack types revealed critical insights into the relative importance of different IMD's components. We found that sensor tampering, particularly with the glucose meter, posed the most significant threat to patient health due to its direct impact on insulin dosing accuracy. In contrast, actuator tampering, such as with the infusion pump, introduced delays that were the least immediately catastrophic. These distinctions provide a nuanced understanding of system vulnerabilities, which can guide the prioritization of defense mechanisms in future designs.

Despite these promising results, several aspects warrant further exploration to solidify the framework's practicality and generalizability. While the proposed RL-based framework demonstrates strong performance, this study is subject to several limitations that open avenues for future research. The experiments relied on a simulated environment to model both the IMD (in this case an insulin pump) and the patient's physiological responses. While this approach offers controlled and repeatable conditions, it may not fully capture the complexity of real-world scenarios, such as unpredictable patient behaviors, environmental factors, hardware limitations, among others. Future work should incorporate real-world testing or more complex physiological models to validate the framework's robustness in practical applications.

The study evaluated a predefined set of attacks, including tampering with sensors, the controller, and the infusion pump. Although these scenarios represent common vulnerabilities, the amount of potential threats is far broader, including network-based attacks, long-term degradation of components, and simultaneous multi-point failures. Expanding the attack repertoire, as well as incorporating more sophisticated attacks, will provide a more comprehensive assessment of the framework's adaptability in future studies.

The RL agent applied a uniform approach to mitigation, resetting components to predefined default states. While effective in this study, real-world IMDs may benefit from more dynamic strategies tailored to specific attack types and patient conditions. Incorporating adaptive strategies based on contextual information, patient history, and risk assessments could enhance both safety and efficiency.

The current framework was tested on an insulin pump system. While the results are promising, it remains to be seen how the approach generalizes to other types of IMDs, such as pacemakers, neurostimulators, or implantable drug delivery systems. Extending the framework to diverse devices will demonstrate its versatility and scalability across the broader field of medical cyber-physical systems. Furthermore, the performance results of our current framework versus other reinforcement learning approaches remain to be seen. As such, this is left as a future direction in our research.

Additionally, different methods, like logistic regression, should also be tested in this problem.

6. Conclusion

In this paper, we proposed a conceptual framework for achieving self-healing in IMDs. Our approach operates in two phases: the first involves a risk assessment network to detect potential threats and evaluate the system vulnerabilities, while the second utilizes a control network trained under the supervision of the risk assessment network to execute corrective actions. Together, these components create a robust system capable of identifying and mitigating risks in real time.

To validate the effectiveness of our self-healing framework, we developed a behavioral abstraction model of an insulin pump as a case study. This model simulated the core functionalities and operational dynamics of a real-world IMD, providing sufficient data to train the risk assessment network and test the control network's ability to respond to adversarial conditions. The results demonstrated the framework's ability to sustain the continuous and reliable operation of the insulin pump, even under a variety of attack scenarios.

By successfully maintaining system stability and ensuring uninterrupted life-critical functionalities, our approach highlights the potential of reinforcement learning when it comes to improving the resilience of IMDs. The insights obtained during this study lay the foundations for future research into self-healing mechanisms, not only for insulin pumps but also for a wider range of medical cyber-physical systems. Moving forward, we hope this methodology could play a pivotal role in improving the safety and reliability of IMDs, ensuring patient health and well-being in increasingly connected and complex healthcare environments.

References

Ahmed, S.F., Alam, M. S.B., Afrin, S., Rafa, S.J., Rafa, N. and Gandomi, A.H. (2024). Insights into Internet of medical things (IoMT): Data fusion, security issues and potential solutions, *Information Fusion* **102**: 102060.

Baker, S.D. (2022). The ironic state of cybersecurity in medical devices, *Biomedical Instrumentation & Technology* **56**(3): 98–101.

Camara, C., Peris-Lopez, P., de Fuentes, J.M. and Marchal, S. (2021). Access control for implantable medical devices, *IEEE Transactions on Emerging Topics in Computing* **9**(3): 1126–1138.

Carreon-Rascon, A.S. and Rozenblit, J.W. (2022). Towards requirements for self-healing as a means of mitigating cyber-intrusions in medical devices, *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Prague, Czech Republic*, pp. 1500–1505.

Chen, X., Zhang, H., Wu, C., Mao, S., Ji, Y. and Bennis, M. (2019). Optimized computation offloading performance in virtual edge computing systems via deep reinforcement learning, *IEEE Internet of Things Journal* **6**(3): 4005–4018.

Deja R., Froelich W., and Deja G. (2021). Mining clinical pathways for daily insulin therapy of diabetic children, *International Journal of Applied Mathematics and Computer Science* **31**(1): 107–121, DOI: 10.34768/amcs-2021-0008.

Dénes-Fazakas, L., Fazakas, G.D., Eigner, G., Kovács, L. and Szilágyi, L. (2024). Review of reinforcement learning-based control algorithms in artificial pancreas systems for diabetes mellitus management, *2024 IEEE 18th International Symposium on Applied Computational Intelligence and Informatics (SACI), Timisoara, Romania*, pp. 565–572.

Dong, X., Hariri, S., Xue, L., Chen, H., Zhang, M., Pavuluri, S. and Rao, S. (2003). Autonomia: An autonomic computing environment, *Conference Proceedings of the 2003 IEEE International Performance, Computing, and Communications Conference, Phoenix, USA*, pp. 61–68.

FDA (2023). Content of premarket submissions for device software functions, US Food & Drug Administration, Rockville, `https://www.fda.gov/regulatory-information/search-fda-guidance-documents/content-premarket-submissions-device-software-functions`.

FDA (2025). Cybersecurity in medical devices: Quality system considerations and content of premarket submissions, US Food & Drug Administration, Rockville, `https://www.fda.gov/regulatory-information/search-fda-guidance-documents/cybersecurity-medical-devices-quality-system-considerations-and-content-premarket-submissions`.

Fox, I., Lee, J., Pop-Busui, R. and Wiens, J. (2020). Deep reinforcement learning for closed-loop blood glucose control, *Proceedings of Machine Learning Research* **2020**(5): 508–536.

Hassija, V., Chamola, V., Bajpai, B.C., Naren and Zeadally, S. (2021). Security issues in implantable medical devices: Fact or fiction?, *Sustainable Cities and Society* **66**: 102552.

HGV Research (2024a). Implantable medical device market size, share & trends analysis report by type, by end use, and segment forecasts, 2024–2030, Horizon Grand View Research, San Francisco, https://www.grandviewresearch.com/horizon/outlook/implantable-medical-device-market-size/global.

HGV Research (2024b). Implantable medical devices market analysis report, Horizon Grand View Research, San Francisco, https://www.grandviewresearch.com/industry-analysis/implantable-medical-devices-market-report.

IBM (2024). Cost of a data breach report 2024, IBM, Armonk, https://www.ibm.com/reports/data-breach.

Johnphill, O., Sadiq, A.S., Al-Obeidat, F., Al-Khateeb, H., Taheir, M.A., Kaiwartya, O. and Ali, M. (2023). Self-healing in cyber–physical systems using machine learning: A critical analysis of theories and tools, *Future Internet* **15**(7), Article no. 244.

Kala, R. (2023). *Autonomous Mobile Robots: Planning, Navigation, and Simulation*, Academic Press, Cambridge, pp. 669–713.

Kegyes, T., Süle, Z. and Abonyi, J. (2021). The applicability of reinforcement learning methods in the development of industry 4.0 applications, *Complexity* **2021**(1): 7179374.

Kingma, D.P. (2014). Adam: A method for stochastic optimization, *arXiv*: 1412.6980.

Koopman, P. (2003). Elements of the self-healing system problem space, *Workshop on Architecting Dependable Systems, Portland, USA*, pp. 31–36.

Kuehn, B.M. (2018). Pacemaker recall highlights security concerns for implantable devices, *Circulation* **138**(15): 1597–1598.

Muhammad, G., Alshehri, F., Karray, F., Saddik, A.E., Alsulaiman, M. and Falk, T.H. (2021). A comprehensive survey on multimodal medical signals fusion for smart healthcare systems, *Information Fusion* **76**: 355–375.

NRC (2001). *Embedded, Everywhere: A Research Agenda for Networked Systems of Embedded Computers*, National Research Council/National Academies Press, Washington, pp. 77–79.

Pirbhulal, S., Zhang, H., Wu, W., Mukhopadhyay, S.C. and Zhang, Y.-T. (2018). Heartbeats based biometric random binary sequences generation to secure wireless body sensor networks, *IEEE Transactions on Biomedical Engineering* **65**(12): 2751–2759.

Pritika, Shanmugam, B. and Azam, S. (2023). Risk assessment of heterogeneous IoMT devices: A review, *Technologies* **11**(1), Article no. 31.

Psaier, H. and Dustdar, S. (2011). A survey on self—healing systems: Approaches and systems, *Computing* **91**(1): 43–73.

Rafajłowicz, W. (2022). *Learning Decision Sequences For Repetitive Processes—Selected Algorithms*, Springer, Cham.

Rao, A., Carreón, N., Lysecky, R. and Rozenblit, J. (2017). Probabilistic threat detection for risk management in cyber-physical medical systems, *IEEE Software* **35**(1): 38–43.

Rathore, H., Mohamed, A. and Guizani, M. (2020). Deep learning-based security schemes for implantable medical devices, *in* A. Mohamed (Ed.), *Energy Efficiency of Medical Devices and Healthcare Applications*, Academic Press, Cambridge, pp. 109–130.

Sallab, A.E., Abdou, M., Perot, E. and Yogamani, S. (2017). Deep reinforcement learning framework for autonomous driving, *arXiv:* 1704.02532.

Seiger, R., Huber, S. and Schlegel, T. (2015). Proteus: An integrated system for process execution in cyber-physical systems, *International Workshop on Business Process Modeling, Development and Support, Stockholm, Sweden*, pp. 265–280, DOI: 10.1007/978-3-319-19237-6_17.

Seiger, R., Huber, S. and Schlegel, T. (2018). Toward an execution system for self-healing workflows in cyber-physical systems, *Software & Systems Modeling* **17**(2): 551–572, DOI: 10.1007/s10270-016-0551-z.

Sutton, R. and Barto, A. (2018). *Reinforcement Learning: An Introduction*, MIT Press, Cambridge.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A., Kaiser, L. and Polosukhin, I. (2017). Attention is all you need, *31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, USA*.

Wang, G., Liu, X., Ying, Z., Yang, G., Chen, Z., Liu, Z., Zhang, M., Yan, H., Lu, Y., Gao, Y., Xue, K., Li, X. and Chen, Y. (2023). Optimized glycemic control of type 2 diabetes with reinforcement learning: A proof-of-concept trial, *Nature Medicine* **29**(10): 2633–2642, DOI: 10.1038/s41591-023-02552-9.

Yu, K.-H., Beam, A.L. and Kohane, I.S. (2018). Artificial intelligence in healthcare, *Nature Biomedical Engineering* **2**(10): 719–731.

Zabihi, Z., Eftekhari Moghadam, A.M. and Rezvani, M.H. (2023). Reinforcement learning methods for computation offloading: A systematic review, *ACM Computing Surveys* **56**(1): 1–41.



Ana S. Carreon-Rascon is an electrical and computer engineering PhD candidate in the University of Arizona. She holds a BS in electronic engineering from the University of Sonora. Her area of focus is medical device security and embedded systems.



Huayu Li is currently an electrical and computer engineering PhD candidate in the University of Arizona. He holds BS and MS degrees in computer and electrical engineering from Northern Arizona University. His current focus is in machine learning, healthcare informatics, digital health, and security.



Jerzy W. Rozenblit is the Raymond J. Oglethorpe Endowed Chair of Electrical and Computer Engineering and a professor of surgery in the College of Medicine at the University of Arizona. He holds PhD and MS degrees in computer science from Wayne State University. His areas of focus are design, analysis, and modeling of complex systems and computer simulations, computer-aided minimally invasive surgery, and applications of computer-based technologies to clinical and academic medicine.



Wojciech Rafajłowicz is a professor in the Wrocław University of Science and Technology. He obtained his PhD from the University of Zielona Góra in 2016 and his DSc from the Częstochowa University of Technology in 2022. His area of research focuses on optimization, optimal control, image processing, and embedded systems. ORCID: 0000-0003-4347-1358

Received: 14 January 2025
Revised: 26 &29 April 2025
Re-revised: 26 June 2025
Accepted: 30 June 2025