| | |
|---|---|
| **Titre:**<br>Title: | PER4Mance Prototyping environment for research on human-machine interactions for alarm floods management: the case study of a chemical plant process control |
| **Auteurs:**<br>Authors: | Karine Ung, Omar Nemer, Aswin Krishna, Moncef Chioua, & Philippe Doyon-Poulin |
| **Date:** | 2022 |
| **Type:** | Communication de conférence / Conference or Workshop Item |
| **Référence:**<br>Citation: | Ung, K., Nemer, O., Krishna, A., Chioua, M., & Doyon-Poulin, P. (octobre 2022). PER4Mance Prototyping environment for research on human-machine interactions for alarm floods management: the case study of a chemical plant process control [Communication écrite]. 66th International Annual Meeting of the Human Factors and Ergonomics Society, Washingon, DC. Publié dans Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 66(1). https://doi.org/10.1177/1071181322661248 |

| | |
|---|---|
| **URL de PolyPublie:**<br>PolyPublie URL: | https://publications.polymtl.ca/10586/ |
| **Version:** | Version finale avant publication / Accepted version<br>Révisé par les pairs / Refereed |
| **Conditions d'utilisation:**<br>Terms of Use: | CC BY-NC-ND |

| | |
|---|---|
| **Nom de la conférence:**<br>Conference Name: | 66th International Annual Meeting of the Human Factors and Ergonomics Society |
| **Date et lieu:**<br>Date and Location: | 2022-10-23 - 2022-10-26, Washington, DC |
| **Maison d'édition:**<br>Publisher: | Sage Publications |
| **URL officiel:**<br>Official URL: | https://doi.org/10.1177/1071181322661248 |
| **Mention légale:**<br>Legal notice: | Ung, K., Nemer, O., Krishna, A., Chioua, M., & Doyon-Poulin, P., PER4Mance Prototyping environment for research on human-machine interactions for alarm floods management: the case study of a chemical plant process control, Proceedings of the Human Factors and Ergonomics Society Annual Meeting (vol. 66, no. 1) pp. 430-434. Copyright © 2022 by Human Factors and Ergonomics Society. doi:10.1177/1071181322661248 |

# PER4Mance Prototyping environment for research on human-machine interactions for alarm floods management: the case study of a chemical plant process control

Karine Ung, Polytechnique Montreal
Omar Nemer, Aswin Krishna, Moncef Chioua, Philippe Doyon-Poulin

Alarm floods are dangerous because the quantity of alarms triggered is too numerous for operators to reliably implement the right corrective action. Process operators of complex systems, such as chemical plants or nuclear power production, are faced with alarm management systems that can be better built in consideration of human capabilities and limitations. Developing human-machine interfaces (HMIs) that better support operators is critical for ensuring the safe and reliable operation of critical systems and processes. The research team has developed an accessible and adaptable prototyping environment dedicated for research on alarm management and human-machine interactions in the process industry. The method used was to build on the Tennessee Eastman Process (TEP) simulator and incorporate Human-Machine design guidelines. The results are an open-sourced prototyping environment that incorporates data from a real chemical plant and integrates true alarm data and thresholds. At the end of this article, we share the Github link to the entire MATLAB, Simulink and App Designer files of PER4Mance: a prototyping environment for research on human-machine interactions for alarm flood management.

## INTRODUCTION

Process alarms play a significant role in maintaining a chemical plant's safety by providing a layer of protection in preventing the occurrence of faults from escalating into process hazards. Alarms aim at helping the process operators keep the plant within normal operating conditions. They provide an indication to the operators that their action is required to fix a fault or to prevent an undesired consequence. Throughout the years, the number and frequency of alarms have increased with technology. In the days of pneumatic controls, installing a new alarm had significant financial costs. The addition of an alarm with mechanical panels required adding light indicators and connecting them (hydraulically) to the sensor. As the number of alarms grew during plant operations, it reached a point where there was no longer any space available on the dashboard to add new alarms (Grimm, 1976). With the use of computer-based control systems, alarms became digital and the operator can configure them by defining thresholds for triggering the alarm (Hollifield & Habibi, 2007). Therefore, adding new alarms no longer had any financial costs or need for additional equipment. Furthermore, with the discovery of each new fault, alarms were added to the alarm system. As a result, the number of alarms has continued to increase over the years to a point where alarms could no longer be handled effectively (Deb & Claudio, 2015). It is common in a process control plant to have well over thousands of alarms per day, a number exceeding the recommended maximum manageable rate of 300 alarms per day (Hollifield & Habibi, 2010).

Detrimental effects of alarm floods on safety and performance are documented in several application domains. In public transportation, automatic train control systems generate alarms to notify train dispatchers of the presence of faulty circuits. The rate of alarms can sometimes reach 8,000 per week and cause the dispatchers to become desensitized to the alarms (NTSB, 2014). In healthcare, the constant alarms from blood pressure machines, ventilators, heart monitors, etc., can cause health professionals to "tune out" the sounds. Alarm desensitization has been highlighted as a widespread problem in hospitals and many alarm-related deaths and injuries have been reported over the past few years (The Joint Commission, 2013). Finally, in the aviation sector, the occurrence of unreliable alarms has shown to foster mistrust and complacency in airline pilots. Studies have shown that alarm-related problems frequently occur across flight operations and that false and incorrect alarms remain a significant concern in aviation (J. P. Bliss, 2003). Research on alarm flood mitigation can be useful in chemical process control as well as across multiple other domains.

According to the Abnormal Situation Management (ASM) Consortium, petrochemical plants suffer one major accident every three years on average (Errington et al., 2009). An important number of these incidents reported were due to poor performance of alarm systems, resulting in plant damages, loss of production, and environmental incidents.

One of the most famous incidents in the field of alarm management is the Milford Haven incident at the Texaco refinery in Pembroke, South Wales, in July of 1994. A massive explosion resulted from 20 tons of flammable hydrocarbons being released from the knock-out pot on the flare header, leading up to hundreds of alarms being triggered. The Health Safety Executive's investigation report (Great Britain & Health and Safety Executive, 1997) identified the concern that alarms can overwhelm the operator, and instead of improving safety, can have the opposite effect and contribute to the incident.

This example illustrates that a fault can affect multiple related systems and trigger an overwhelming number of alarms. An alarm flood is defined as 10 or more annunciated alarms in a 10-minute period per operator (ANSI/API, 2010). In ISA-18.2 it is stated as: "A condition during which the alarm rate is greater than the operator can effectively manage (International Society of Automation, 2022a).”

Alarm floods are troublesome because the quantity of alarms triggered is too numerous for operators to manage, making it difficult to implement the right corrective action. A fault can lead to a cascade of alarms, or multiple faults can occur during the same time period. Both scenarios can lead to an alarm flood, without any alarm differentiation between the separate faults. This phenomenon can affect hundreds or even thousands of alarms, with many unnecessary and redundant alarms resulting from the same root cause being enunciated and displayed to the operator. The discrepancy between the amount of information presented and the amount of information to which individuals can effectively manage leads to increased workloads, human error, and decreases in efficiency (Stanton et al., 2009). Despite improvements in alarm rationalization and prioritization processes, alarm floods are still a significant issue in abnormal situation management (Parsa et al., 2022). In alarm flood situations, one of the only responses available to the operators is to acknowledge and silence the alarms (Bransby & Jenkinson, 1998).

ANSI/ISA-18.2 Management of Alarm Systems for the Process Industries and the EEMUA 191 Human-Machine Interfaces (HMI) are standards providing guidelines for alarm systems management in process control plants. However, alarm systems built using these standards still need to be tested in a safe environment with human operators prior to the implementation in real operating industrial processes (Goel et al., 2017).

There are existing prototyping tools or simulation environments available for HMI test, but with limited availability. For instance, the company Corys (Corys, 2015), provides high-fidelity and dynamic simulators. Their simulator has been previously used in a human-in-the-loop study which investigated the impact of alarm management system design i.e. alarm rationalization, on the process operator’s workload (Simonson et al., 2022). However, the simulator comes at a financial cost that limits its accessibility to the public. Other researchers code their own simulator (Long et al., 2022), but their simulator and its codes are not made available to the general public.

Alarm systems designed according to safety considerations provide the primary source of warning for operators when it comes to abnormal situations. Still, to the best of our knowledge, there has been no freely available and open-source process control simulator environment that has been developed to provide a platform for research on human-machine interactions during alarm floods.

Following the approach of Simonson et al. 2022, we developed a human-machine prototyping environment that can be used as a research tool to investigate alarm flood management in a process control environment. We aimed at creating an environment that can promote the study of the impact of machine learning-based decision support systems to guide the operator during periods of alarm floods, what we’ll call the "diagnostic tool". The next section presents the development method, followed by validation results, discussion and conclusion.

## METHOD

### Step 1 - Tennessee Eastman Process (TEP)

The first step in creating the prototyping environment was to use the Tennessee Eastman Process (TEP) simulator to represent a chemical process control (Chen, 2019). The TEP is a realistic simulation of a chemical process that runs on MATLAB (The MathWorks Inc., 2021). It consists of five main process units: a reactor, a separator, a stripper, a compressor and a condenser (Figure 1).

The process has a total of eight different chemical components identified as A through H. These components consist of three gaseous reactants, A, D, and E that are fed to the reactor, which contains a small amount of inert gas B. There is also the gaseous reactant C that is fed directly into the stripper. Liquid products G and H exit the stripper base and are transferred to subsequent units and cells. The primary objectives of the process are to maintain the specified ratio of G/H in the product and maintain the specified product rate during normal operation and process disturbances. There is also a liquid by-product F which is purged from the TEP. The operator can manipulate 12 input variables and monitor 41 output variables. The TEP simulator also has 20 pre-defined fault scenarios (Bathelt et al., 2015). The process control community has used TEP extensively as a benchmark to compare the performance of control strategies, but has received little attention as a user-facing simulator (Udugama et al., 2020).

### Step 2 - TEP alarm dataset

As the TEP simulator did not comprise of alarms embedded in its program, the second step of the tool development consisted of adding an alarm dataset to the prototyping environment. We used the work from the IEEE TEP Alarm Management Dataset (Manca, Gianluca, 2020), where the authors identified the TEP variables with their alarm high and low threshold values. We programmed their alarm thresholds into our tool, so that the alarms are triggered at the correct threshold limits. Therefore, whenever a variable’s actual value crosses the high or low threshold, the respective alarm is triggered.

### Step 3 - Real-Time Data Exchange

The next step involved creating a real-time data exchange link between the TEP simulator and our prototype. By adding a scope block, the prototyping environment is able to locate the variables and read the data from Simulink (The MathWorks Inc., 2021). We added a single scope block to the default configuration of the TEP at the output block of the variables. This enabled us to read the data of the variable outputs from our prototype. Furthermore, by adding the additional scope blocks to all the variables, we managed to capture the data generated by the simulator during its execution, and were able to display them in real-time on our prototype. In addition to reading the data, this also allowed us to make input changes to the variables during the simulation. It was therefore possible for operators to change the manipulated variables, i.e. the valves opening and setpoints, while the environment was running.

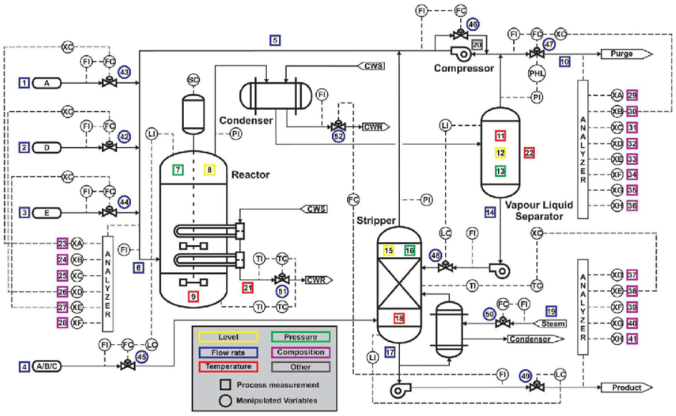


Figure 1: Piping and Instrumentation Diagram of the Tennessee Eastman Process (Ma et al., 2020)

**Step 4 - Human-Machine Guidelines**

Having the TEP simulator with its alarm dataset defined and the data exchange established, we were ready to design the tool's interface on MATLAB App Designer. We followed human-machines guidelines of ISA-101 which provides a design model called High-Performance (HP) HMI to design an interface that allows operators to detect, diagnose and correct efficiently dynamic operations in a process control environment (International Society of Automation, 2022b). More specifically, the standard provides guidelines on how to display information when developing the prototype's interfaces.

The presentation of the data should be done in a hierarchical manner across four levels. The first level is the most important and should present a global view of the whole process. It is also where information about the most critical equipment should be displayed. The second level is dedicated to the subunits of the system, with each subunit having its own view presenting more detailed information on its operating conditions than on the first level. The third level is an even more specific view of a particular piece of equipment of a subunit. Finally, the fourth level contains any other useful information that can help the operators make their diagnosis. For levels one and two, we identified the tasks the operators need to be able to perform, and defined the relevant variables. We omitted levels three and four because all the information that was identified as relevant during our analysis could be transmitted within the first two levels. Then, we defined the format for each variable (e.g., graphs, trends, thermometers, lists, etc.) depending on their context.

Following these HP HMI principles, we were able to identify where to present the 41 variables and their format, the key performance indicators (KPIs), the alarms, the diagnostic tool and the controllers for the manipulated variables. We will present them in the following section.

**Step 5 - Implementation**

We chose to use App Designer (The MathWorks Inc., 2022) as the development tool because it is an extension of Matlab, which was required for the TEP simulator to function properly. Since these three modules are under the same working environment, communication and data exchange was running properly. Moreover, the App Designer tool offers a library of objects (graphs, gauges, etc.) ready to use that can be dragged and dropped onto the interface.

## RESULTS

The prototyping environment consists of two interfaces, one interface open per computer monitor simultaneously. The first interface (Figure 2) represents the system overview, containing the global and critical information showing the system's health status (level 1). This overview interface has a panel on the top that provides the key performance indicators of the system. These are the inputs' flow rate, their concentration to the reactor, production rate, quality of the G and H products, production cost per hour and finally the concentration of the chemical components at the output, including the purge and the products. In this same section, on the right, we have the diagnostic tool which displays a solution when a fault occurs. The user of the prototype can choose to provide a correct solution, an incorrect solution or no solution at all. In the middle section, we integrated a diagram representing the logical flow of the TEP system from left to right so that the operators have a global view of the process.

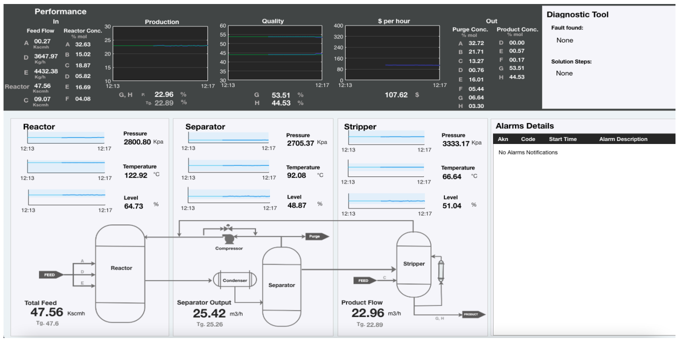


Figure 2 - The environment system overview

These three blocks show the most critical equipment, i.e. the reactor, the separator and the stripper. For each of these equipment, we displayed its pressure, temperature and flow. At the bottom of the screen are three digital readouts indicating the incoming flow to the reactor, the outgoing flow from the separator and the overall flow of products for the stripper. To the right of these blocks is the alarm table. Under normal circumstances, there are no alarms displayed.

The second interface (Figure 3) represents detailed information per unit, displaying the variables related to the reactor, condenser, separator, compressor and stripper under different tabs (level 2). On the right side, the users can act on the process valves either in manual mode (openings adjustments) or in automatic mode (setpoint settings). If the control is in automatic mode, the operator can modify these setpoints. If the control is in manual mode, the operator can directly modify the valve opening.

At the top of the screen, there are tabs to navigate to other units of the system. The units on these tabs follow the process flow navigation from left to right. Some units are simpler than others, therefore, we combined them to save screen real estate; the condenser, separator and purge; and the stripper with the final product information.

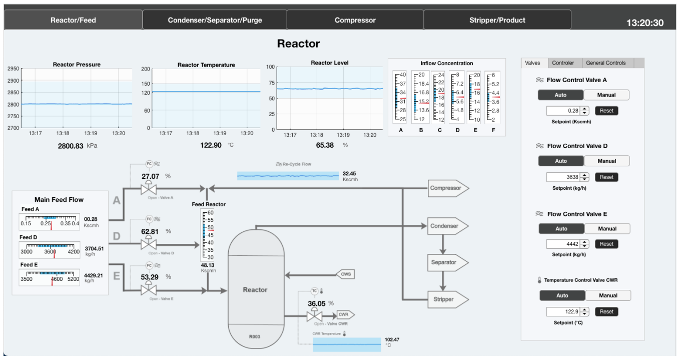


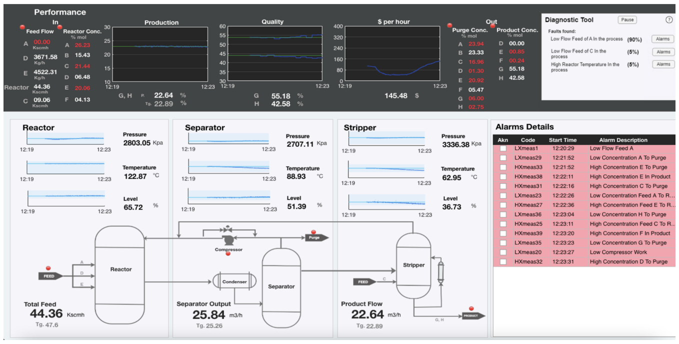
Figure 3 - The reactor interface

The interface adds further information when the system is under a fault condition. Figure 4 presents a fault condition scenario: a loss of feed of input A. The red dots indicate the variables and units have exceeded their normal threshold values and that an alarm has been triggered. They can be seen on the top KPIs section, but also on the overall TEP diagram. Furthermore, there is the alarm table on the bottom right side of the interface. There is a checkbox on each alarm line so that the operators can indicate that they have acknowledged the alarm. There is also the code of the alarm, its time of appearance, its description and finally the unit involved.

The prototyping environment follows the standards of a High Performance HMI: a two-level hierarchy was used and the data was grouped according to their corresponding sub-unit. Trend graphs and analog indicators were used to visualize if the value of a variable is within the normal range. The number of colors were limited by keeping the background gray, the operating limits in blue and the fault indications in red. By following these standards, the simulator environment closely resembles the interfaces used in the industry, and users have access to a functioning prototype that has an interface that represents those used in real-life-operations.

## DISCUSSION

Although there are multiple process control simulator environments available in the market, not all are accessible and malleable. Our prototyping environment differs from others in the following ways: it has no financial costs, it is open-sourced and it is extremely modifiable. From our original files, users can change the codes and interfaces freely.



Diagnostic Tool

Faults found:

| | | |
|---|---|---|
| Low Flow Feed of A In the process | (90%) | Alarms |
| Low Flow Feed of C In the process | (5%) | Alarms |
| High Reactor Temperature In the process | (5%) | Alarms |

Alarms Details

| Akn | Code | Start Time | Alarm Description |
|---|---|---|---|
| ☐ | LXmeas1 | 12:20:29 | Low Flow Feed A |
| ☐ | LXmeas29 | 12:21:52 | Low Concentration A To Purge |
| ☐ | HXmeas33 | 12:21:52 | High Concentration E To Purge |
| ☐ | HXmeas38 | 12:22:11 | High Concentration E In Product |
| ☐ | HXmeas31 | 12:22:16 | High Concentration C To Purge |
| ☐ | LXmeas23 | 12:22:26 | Low Concentration Feed A To R... |
| ☐ | HXmeas27 | 12:22:36 | High Concentration Feed E To R... |
| ☐ | LXmeas36 | 12:23:04 | Low Concentration H To Purge |
| ☐ | HXmeas25 | 12:23:11 | High Concentration Feed C To R... |
| ☐ | HXmeas39 | 12:23:20 | High Concentration F In Product |
| ☐ | LXmeas35 | 12:23:23 | Low Concentration G To Purge |
| ☐ | LXmeas20 | 12:23:27 | Low Compressor Work |
| ☐ | HXmeas32 | 12:23:31 | High Concentration D To Purge |

Figure 4 - Abnormal condition: loss of flow A

While developing this prototyping environment, there were a few limitations encountered. First, App Designer offered a limited library of graphical elements. Although the tool is very easy to use, the graphical elements provided by the program looked out-dated. The second disadvantage of this tool is that the more we added graphical elements to the interface, the heavier and slower the editing mode became. Finally, it is worth mentioning that the running speed to complete the simulation was less than 1 minute. The prototyping environment reflects the same speed as Simulink, and we therefore had to slow down the running speed on Simulink to be able to have a working prototype.

Despite these limitations, users can change the alarm thresholds and behaviors, add or remove alarms, and configure the information provided by the diagnostic tool. Furthermore, users can remove the input controllers in case they want to reduce operator's control over the environment. We also developed this prototype to allow research with one operator, but modifications of the environment to study multiple operators simultaneously or team dynamics could be possible and worth investigating. We've created this prototyping environment to reflect our own goal, which is to study the impact of machine learning-based decision support systems, the diagnostic tool, to guide the operator during periods of alarm floods. But this prototype can be modified and adapted to countless other environments to study different aspects of alarm management in industrial settings.

## CONCLUSION

This paper aimed to address the need for a prototyping environment to study human-machine in the process industry. We've proposed a prototyping environment that was built on the TEP simulator and HMI design guidelines and principles. With this public tool, we hope to encourage shared research on human-machine interaction and alarm management in relation to machine learning systems.

The next step for the research team is to test the prototyping environment with humans. Testing with humans will allow us to improve the prototype itself and to study the effects of the prototype's design on human cognition. Even though we followed HMI design principles, there are still many elements in the prototyping environment itself that can be improved, such as the way the variables are presented, the alarm presentation and the solutions conveyed by the diagnosis box. Also, testing the fault scenarios of the prototype with humans will allow us to perform fundamental research on alarm flood management and diagnosis, thereby investigating the interface elements that would help or hinder human diagnosis abilities. We have made this prototyping environment available to all in order to encourage shared learning and promote further work on improving the prototype.

The US Federal definition of research is "a systematic investigation, including development, testing, and. evaluation, designed to develop or contribute to generalizable knowledge" (45CFR46.102). We hope that our approach and design might generalize to other research in simulations or alarm floods in different domains. Don't hesitate to contact us for any collaborative work.

The PER4Mance (MATLAB, Simulink and App Designer files) are available to download via the following link: https://github.com/karine-ung/perf4mance

## REFERENCES

ANSI/API. (2010). *Process Safety Performance Indicators for the Refining and Petrochemical Industries, First Edition*. American Petroleum Institute.

Bathelt, A., Ricker, N. L., & Jelali, M. (2015). Revision of the Tennessee Eastman Process Model. *IFAC-PapersOnLine*, *48*(8), 309–314. https://doi.org/10.1016/j.ifacol.2015.08.199

Bransby, M. L., & Jenkinson, J. (1998). *The management of alarm systems*. HSE Books.

Chen, X. (2019). *Tennessee Eastman simulation dataset* [Data set]. IEEE DataPort. https://doi.org/10.21227/4519-Z502

Corys. (2015). https://www.corys.com/en/simulator-range

Deb, S., & Claudio, D. (2015). Alarm fatigue and its influence on staff performance. *IIE Transactions on Healthcare Systems Engineering*, *5*(3), 183–196. https://doi.org/10.1080/19488300.2015.1062065

Errington, J., Reising, D. V., Burns, C., & ASM Joint R & D Consortium. (2009). *Effective alarm management practices*. ASMConsortium.

Goel, P., Datta, A., & Mannan, M. S. (2017). Industrial alarm systems: Challenges and opportunities. *Journal of Loss Prevention in the Process Industries*, *50*, 23–36. https://doi.org/10.1016/j.jlp.2017.09.001

Great Britain & Health and Safety Executive. (1997). *The explosion and fires at the Texaco refinery, Milford Haven, 24 July 1994: A report of the investigation by the Health and Safety Executive into the explosion and fires on the Pembroke Cracking Company Plant at the Texaco Refinery, Milford Haven on 24 July 1994.* HSE Books.

Hollifield, B. R., & Habibi, E. (2007). *Alarm management: Seven effective methods for optimum performance*. Instrumentation, Systems, and Automation Society. http://app.knovel.com/hotlink/toc/id:kpAMSEMOP1/alarm-management-seven

Hollifield, B. R., & Habibi, E. (2010). *The alarm management handbook: A comprehensive guide : practical and proven methods to optimize the performance of alarm management systems*.

International Society of Automation. (2022a). *ANSI/ISA 18.2 Management of Alarm Systems for the Process Industries*. https://www.isa.org/intech-home/2016/may-june/departments/isa18-alarm-management-standard-updated

International Society of Automation. (2022b). *ISA101 Human-Machine Interfaces*. https://www.isa.org/standards-and-publications/isa-standards/isa-standards-committees/isa101

Long, S. K., Lee, J., Yamani, Y., Unverricht, J., & Itoh, M. (2022). Does automation trust evolve from a leap of faith? An analysis using a reprogrammed pasteurizer simulation task. *Applied Ergonomics*, *100*, 103674. https://doi.org/10.1016/j.apergo.2021.103674

Ma, L., Dong, J., & Peng, K. (2020). A novel key performance indicator oriented hierarchical monitoring and propagation path identification framework for complex industrial processes. *ISA Transactions*, *96*, 1–13. https://doi.org/10.1016/j.isatra.2019.06.004

Manca, Gianluca. (2020). *"Tennessee-Eastman-Process" Alarm Management Case Study* [Data set]. IEEE DataPort. https://doi.org/10.21227/326K-QR90

Parsa, K., Hassall, M., & Naderpour, M. (2022). Enhancing Alarm Prioritization in the Alarm Management Lifecycle. *IEEE Access*, *10*, 99–111. https://doi.org/10.1109/ACCESS.2021.3137865

Simonson, R. J., Keebler, J. R., Blickensderfer, E. L., & Besuijen, R. (2022). Impact of alarm management and automation on abnormal operations: A human-in-the-loop simulation study. *Applied Ergonomics*, *100*, 103670. https://doi.org/10.1016/j.apergo.2021.103670

The Joint Commission. (2013). Medical device alarm safety in hospitals. *Sentinel Event Alert*, *50*. https://www.pwrnewmedia.com/2013/joint_commission/medical_alarm_safety/downloads/SEA_50_alarms.pdf

The MathWorks Inc. (2022). *App Designer: Create desktop and web apps in MATLAB*. https://www.mathworks.com/products/matlab/app-designer.html

Udugama, I. A., Gernaey, K. V., Taube, M. A., & Bayer, C. (2020). A novel use for an old problem: The Tennessee Eastman challenge process as an activating teaching tool. *Education for Chemical Engineers*, *30*, 20–31. https://doi.org/10.1016/j.ece.2019.09.002

**Karine Ung**
Polytechnique Montréal
Département de mathématiques et de génie industriel

**Omar Nemer**
Polytechnique Montréal
Département de génie informatique et génie logiciel

**Aswin Krishna**
Indian Institute of Technology, Guwahati

**Moncef Chioua**
Polytechnique Montréal
Département de génie chimique

**Philippe Doyon-Poulin**
Polytechnique Montréal
Département de mathématiques et de génie industriel