Workshop No. 2
Dynamical Systems Analysis & Design

Carlos Andres Celis Herrera - 20222020051

Xiomara Salomme Arias Arias - 20222020028

Universidad Distrital Francisco Jose de Caldas

System Sciences Foundations

Carlos Andres Sierra VirgÜes

May 12th, 2025

Dynamical Systems Analysis & Design

The objective of this workshop is to incorporate methods from systems science and cybernetic theory into the autonomous agent, understanding real-time changes, feedback loops and nonlinear behaviors.

Therefore, we made some changes to the system diagram (Figure 1). We adjusted the feedbacks and incorporated new adaptive logic for Insight Engine, also, we add external actor (specialist) who helps to supervise and evaluate the recomendations of the system. In addition, we added new inputs that are very important for improving the walker recommendations, such as training data and the patient's medical history.

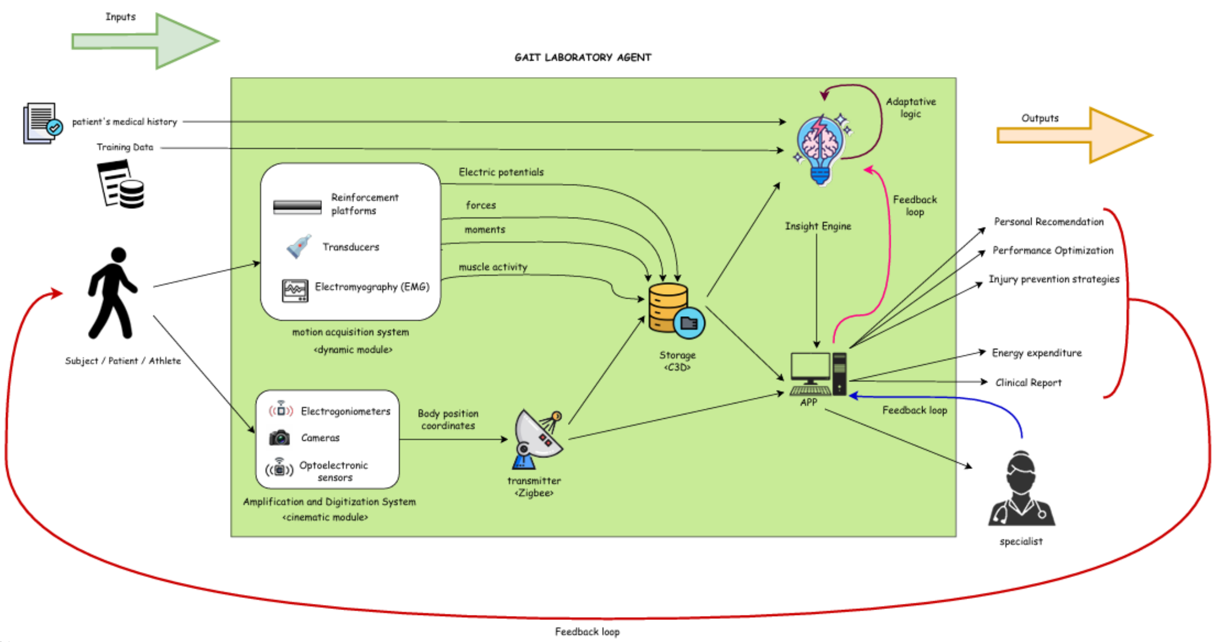


Figure 1

Gait Laboratory Diagram

System Dynamics Analysis

Causal Model as a Basis for Simulation

The causal model developed represents a complex sensor-assisted gait assessment system (Figure 2), in which key components such as data collection, storage, processing by an inference engine (Insight Engine) and feedback to the patient through personalized recommendations are identified.

This model allows the identification of feedback loops, nonlinear relationships and temporal dependencies, which helps to analyze their evolution over time, even without detailed quantitative simulation.

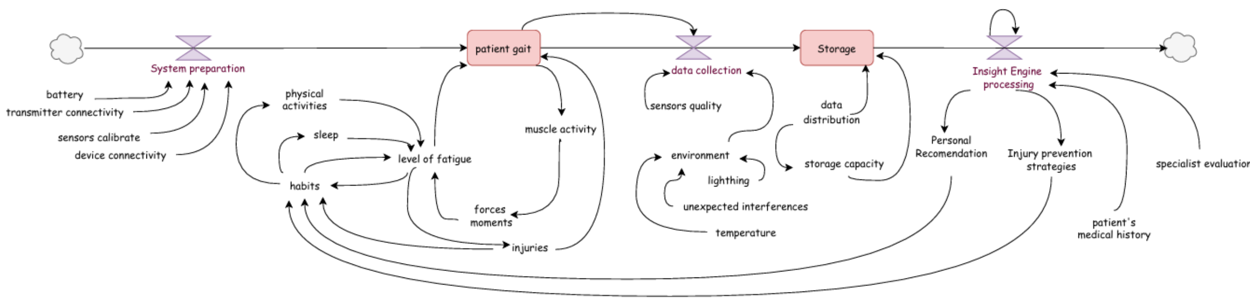


Figure 2

Causal Diagram

To understand the diagram properly, it is important to recognize the symbols that compose it. The opposing triangles (in the shape of a valve) represent activities or processes that mark a transition between stages of the system. The squares represent stocks, meaning elements that accumulate or maintain a state over time. Arrows indicate relationships of influence: they point from the variable that has the effect to the variable that receives it.

Non-linear Factors

The following is a list of variables or relationships in the model that show non-linear behavior, which means that their effect is not proportional to their cause:

- **Fatigue vs. Patient Gait:** The relationship between fatigue and patient gait quality presents a clear non-linearity, as the impact of fatigue is not proportional in all phases of exercise or rehabilitation. It should also be taken into account that factors such as the patient's habits, sleep quality and previous activities influence the way patient gaits. Initially, small increases in fatigue may have minimal effects on gait; however, once a certain physiological threshold is crossed, the deterioration in gait quality rapidly intensifies. This behavior implies that the system must be especially sensitive to detect early signs of fatigue, as a delay in intervention can trigger disproportionately negative consequences, such as loss of stability, risk of falls or aggravation of previous injuries.

- **Environment vs. Data Collection:** The relationship between environmental conditions and data capture accuracy is clearly non-linear, as small changes in factors such as lighting, temperature, unexpected interference or environmental stability may not have an immediate impact on the sensors; however, when certain critical limits are exceeded, data quality degrades abruptly and significantly. For example, a small drop in illumination can be tolerated by optical sensors, but a more pronounced drop can lead to inaccurate readings or outright loss of data.

- **Level of Personal Recomendations vs. Effectiveness of the intervention:** This relationship represents a nonlinear learning curve, where at first, more general adjustments may have limited impact. However, as the system collects more patient-specific data (such as medical history, treatment responses, training data), smaller changes in personalization, made by the Insight Engine and specialist assessment, can generate much more noticeable improvements in efficacy. This reflects the fact that personalization does not always have an immediate impact, but once the system begins to understand the patient better, the return on that personalization increases exponentially.

Time-Dependent Factors

They are those that change over time and whose evolution influences the behavior of the system.

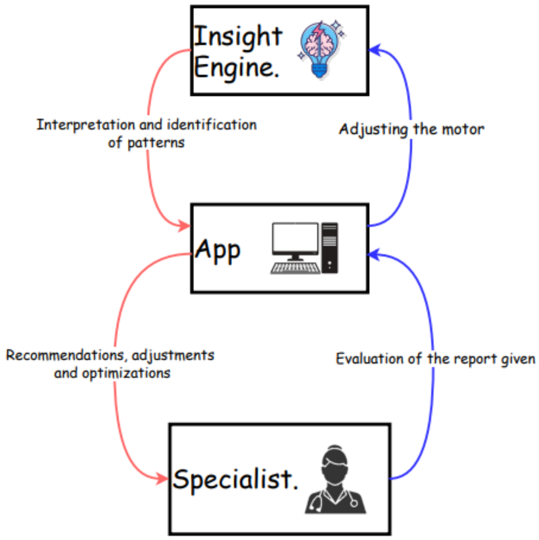
- Level of fatigue: Increases or decreases according to sleep, physical activity and habits, and changes constantly with the patient’s condition.

- Muscle activity: Varies over time depending on physical condition, injuries and gait quality.

- Patient gait: Evolves progressively with the patient's condition, fatigue, injuries and recommendations.
- Injuries: Accumulates or decreases over time depending on activity and effectiveness of recommendations.
- Storage capacity: Changes dynamically as more data is collected and processed.
- Sensor quality: May degrade with use or change due to maintenance/calibration over time.
- Environment: Although external, its conditions (temperature, light, interference) are constantly changing and impact the system.
- Data collection: Occurs continuously during monitoring, affecting system dynamics.
- Personal Recommendation: Changes over time according to new data, feedback and system learning.
- Insight Engine processing: Evolves according to the volume and quality of data received and accumulated personalization.

Feedback Loop Refinement

The system now has two new feedback cycles. The first one is present in the Insight engine, in which it adjusts itself based on the data it processes and the results it generates to the point where it can be considered as self-learning by the element. On the other hand, the second cycle consists of a signal given by an external element considered a specialist, indicating how good the results of the Insight engine are. This last cycle can be considered a reward signal that indicates precisely and in real time how good a recommendation was in a specific case.



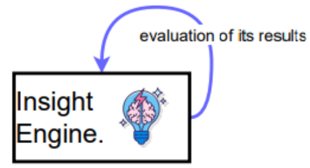


Figure 3

Diagrams of the new feedback loops

Stability and Convergence One of these is the BIBO (Bounded Input, Bounded Output) stability, which refers to the fact that any bounded input expects a similarly bounded output. That is to say, in our agent we expect recommendations focused on physical aspects, since the input of the system is mostly physical data of the patient. On the other hand, the motor of the system agent starts being trained, so experience, time and reward signals present a convergent behavior since its outputs will be stabilizing as it learns and its number of errors decreases. In the same way, if the physical results of the patients improve thanks to the recommendations of the system, this indicates that the outputs of the system are stabilizing.

Iterative Design Outline

Dates Structures

Data is an essential part of the project, from the use for motor training, processing and gait analysis. Therefore, a good use of this data leads to fulfill the objective of the system. With this in mind, the structure of our data is as follows.

- Physical data

Physical data						
Electric potentials	Forces	Moments	Muscle activity	position	Weight	height

- Medical history

Medical history			
AgePacient	injuries	laterality	Constraints

- Experiences Buffer

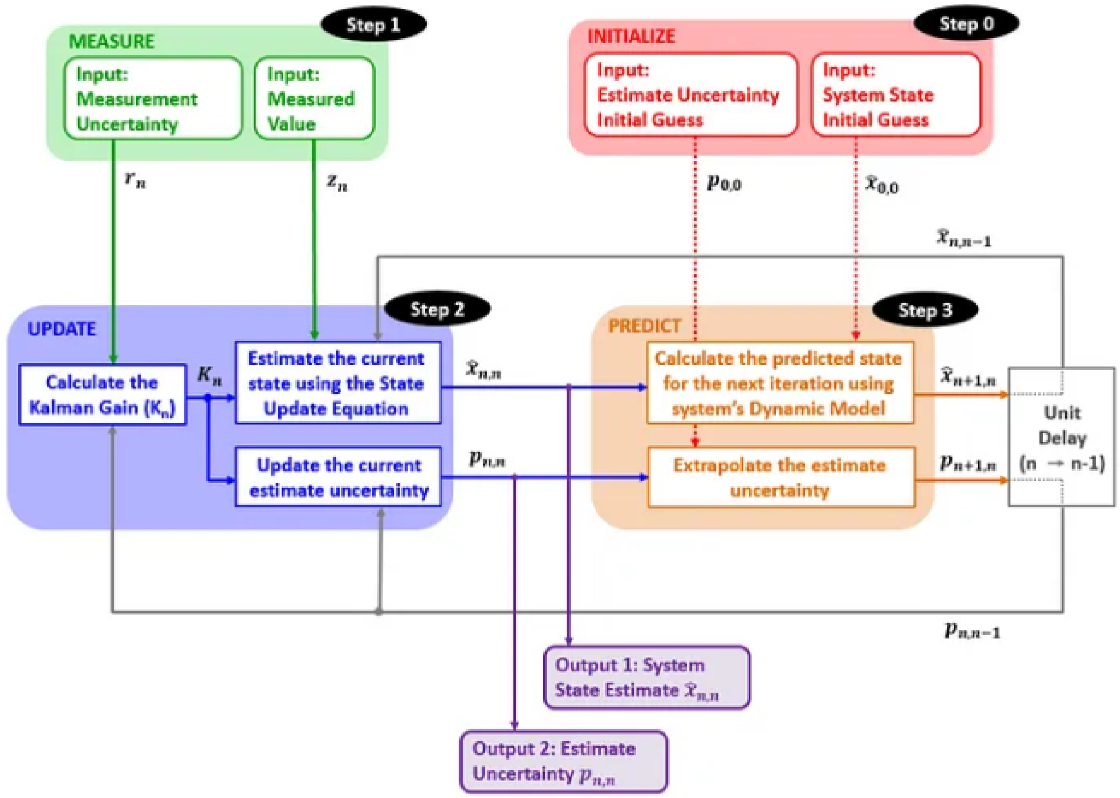
Experiences Buffer			
Experience	datosCuadros	feedbackLoop	Predictions

On the other hand, these system data structures can be stored in hash tables in order to efficiently store and retrieve data. The main idea of hash tables is to associate keys or keys with values by means of a hash function. A hash function is a mathematical operation that converts any input into a fixed-length output, called a hash or hash code. The hash code is used as an index to store and access the value in an array, called a bucket.

Algorithms

- **Kalmar Filter Algorithm:** The Kalman filter is a recursive estimation algorithm that improves the prediction of hidden or noisy variables from a combination of theoretical models and empirical measurements. In the context of sports science, this filter has proven to be a powerful tool for estimating variables such as fitness, fatigue or motor performance from incomplete or noise-affected data, such as from portable sensors (IMU, GPS, EMG). According to Pathak (2021), the usefulness of the Kalman filter lies in its ability to optimally integrate prior knowledge of the system (dynamic model) with noisy observations of the environment, adjusting in real time the prediction of the state of the athlete or biomechanical system. This makes it an ideal resource for intelligent agents that need to adapt their decisions to the evolution of the user's internal state, such as in gait laboratories.

  Figure 4 shows the diagram of the Kalman Filter algorithm, it is very useful to understand how to implement it in the context of the March Lab.



Detailed description of the Kalman Filter algorithm

Figure 4

Detailed description of the Kalman Filter algorithm

- **Deep Q-Learning (DQN):** The Deep Q-Learning (DQN) algorithm is an extension of classical Q-learning that uses deep neural networks to approximate the action-value (Q) function, allowing it to scale to environments with continuous and complex state spaces. Instead of storing a table of Q-values for each state and action, DQN employs a neural network that predicts the Q-values for all possible actions given the observed states. This technique has been key in successful applications of reinforcement learning, such as the control of autonomous agents in video games or recommender systems.

  In the context of an adaptive agent in a gait laboratory, DQN could allow the system to learn, by experience, when and how to optimally intervene in a patient's gait, based on the perceived biomechanical state and reward signals defined by improvements in symmetry, efficiency or stability. Thus, the agent could develop a personalized adaptive policy that improves over time, even in the presence of variability between users or clinical conditions Kukreja (2020).

Frameworks

To dynamically model the patient's gait system, a framework can be implemented that combines **SimPy**, a discrete event simulation library, with **Gymnasium**, a reinforcement learning environment. In this scheme, **SimPy** would be in charge of simulating the passage of time and the evolution of variables such as fatigue, rest or the quality of recommendations, making it possible to represent nonlinear and time-dependent behaviors. For its part, **Gymnasium** defines the decision-making environment of the agent (e.g., the personalized recommendation system or the analysis engine), which takes actions based on observations of the simulated environment and receives rewards based on the effectiveness of the intervention or the improvement in the patient's progress. This integration makes it possible to train adaptive agents under realistic and stochastic conditions, evaluating policies that optimize the system's long-term performance.

Dynamic test

- **Simulation parameters**: The simulation of the system begins with a series of sensors that measure different aspects of gait, some of these are: optoelectronic sensors, electrogoniometer sensors, reinforcement platforms, transducers, among others. By means of these important physical parameters of the patient's gait are measured, these should be the next to be simulated, since they are the input of the system and with which a subsequent analysis will be made to determine the patient's condition.

- **Scenario variations**: They are different types of scenarios in which parameters change between them, but without losing the central context. For example: User with knee injury, his gait will not be the best and the system will generate more recommendations; Healthy and rested user, his gait will be more controlled and he will not get tired so much, the system may not give many recommendations. As well as these two particular cases, there are many more scenarios where the mere fact of changing a parameter such as: increasing the weight, speed or time during walking, generates different scenarios to analyze but without losing the context that is to optimize the gait.

References

Kukreja, R. (2020, August 5). *Diving into deep reinforcement learning with deep q-learning.* Retrieved from https://towardsdatascience.com/diving-into-deep-reinforcement-learning-with-deep-q-learning-376e588bb803 (Towards Data Science)

Pathak, S. (2021). *Kalman filter in sport science.* Retrieved from https://swetankpathak.medium.com/kalman-filter-in-sport-science-3df199d10d5b (Accessed: 2025-05-08)