Portfolio

MSc Graduation Projects

Bayesian Intelligent Autonomous Systems lab

March 11, 2022

1 Cocktail party problem

Problem description

You are challenged to design an agent that learns to solve the cocktail party problem through on-the-spot interactions with a (human) listener. The cocktail party problem refers to the issue of not being able to understand your conversation partner in the presence of many simultaneously competing voices (Fig.1). The learning protocol is displayed in Fig.2. A listener wears earbuds that are capable to process audio signals in real-time (like hearing aids). In response to a detected problem, the agent proposes the most promising alternative parameter settings for the audio algorithm (the TRY step). Next, the new audio algorithm is executed in the ear buds and evaluated by the listener (EXECUTE and EVALUATE steps). Based on the listener's appraisal, the agent should now update its model of the world (LEARN step). This design loop repeats in real-time until the listener indicates that the problem has been solved.

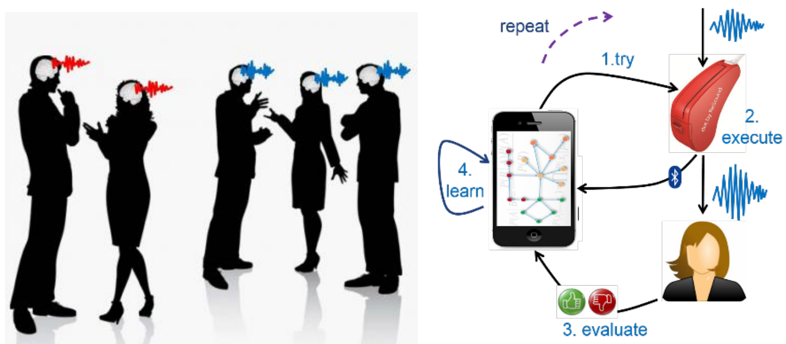

Figure 1: (Left) Cocktail party. (Right) Hearing aid.

Student task description

This project will get you involved with the latest artificial intelligence methods, since the agent needs to (1) learn from each interaction and (2) be smart about selecting the most promising algorithm candidates. It will also give you an opportunity to learn about how biological brains solve real-time design issues.

Concrete tasks:

- Familiarize yourself with the literature on Active inference.
- Discuss problem and solution proposal with researchers at BIASlab.
- Implement solution proposal using BIASlab resources.
- Experiment with the implementation and compare to current solutions.

- Analyze results and reflect on what has been achieved.
- Write a report detailing the advantages and limitations of this approach.

Project positioning

The Bayesian Intelligent Autonomous Systems lab BIASlab (http://biaslab.org, FLUX-7.060) is a subgroup of the Signal Processing Systems (SPS) that aims to develop Intelligent Autonomous Agents. These agents interact with their environment through their sensors and actuators in order to learn purposeful behavior, e.g., to navigate, play soccer or they may learn to decode speech signals under bad acoustic conditions. Our research projects are inspired by the latest insights from machine learning, computational neuroscience and signal processing.

Supervision:

- Weekly progress meetings with Bert de Vries.
- Periodic progress meetings with contacts at GN Resound.
- The student should prepare update meetings, preferably with derivations or visualizations in interactive notebooks.

Timeline

The project is available from September 2017 onwards. The total duration will be 32 weeks. Halfway through, there will be a ”midterm” evaluation where the student must report on their activities and indicate how they will proceed for the remainder of the project. At the end, the student will write a thesis in 10-page double-column IEEE Transactions style summarizing their work, their findings and possible future steps. The thesis will be presented in an official ”defense” ceremony and a committee of experts will grade the student’s work.

2 Active inference for SpotMicro locomotion

Problem description

The goal is to develop an intelligent autonomous system (agent) for a SpotMicro quadruped robot (see Fig. 2 left). The agent must learn to walk: it will not be given an accurate model of its legs' kinematics but will have to gradually build a locomotion model from interaction with its environment. You will use *Active Inference* (AIF), a probabilistic machine learning framework from the computational neuroscience community, to design and train the agent.

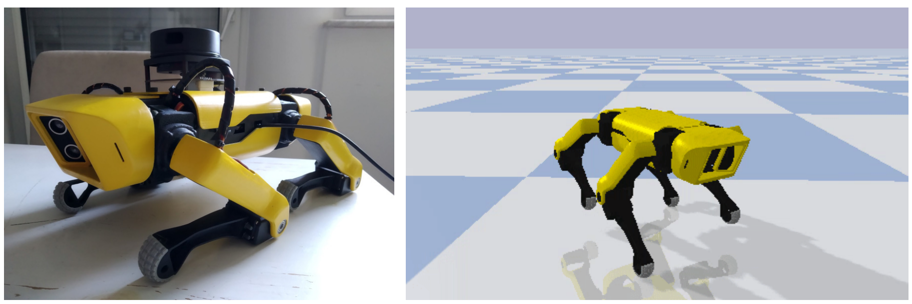

Figure 2: (Left) A physical SpotMicro. (Right) A SpotMicro in simulation. Figures courtesy of `https://spotmicroai.readthedocs.io/en/latest/`.

Please note that this is not a hardware project, but an AI project. You will write code to control a simulated robot that interacts with a simulated environment (see Fig. 2 right), generated using the open-source physics engine Bullet. The challenge will be to adapt existing AIF agents to control a SpotMicro system.

Project positioning

This is a purely academic graduation project. You will be working in the Bayesian Intelligent Autonomous Systems lab (BIASlab), a subgroup of the Signal Processing Systems (SPS) group at TU Eindhoven. BIASlab develops probabilistic models and inference algorithms for signal processing and control problems. This graduation project is a part of a larger research initiative towards AIF agents for robot locomotion.

Student task description

You will initially be spending time familiarizing yourself with the tools and techniques that BIASlab develops (mainly ReactiveMP.jl). Once familiar, you will write your own active inference agent based off of existing AIF agent implementations within BIASlab. Note that you'll be supported by BIASlab researchers that are working on other robot locomotion projects.

Concrete tasks

- Read the literature on AIF agents for control.
- Familiarize yourself with the challenges of quadrupedal locomotion.
- Learn to use BIASlab's probabilistic machine learning toolbox *ReactiveMP*.
- Familiarize yourself with the sensors and actuators on-board a SpotMicro and the simulation environment Bullet.
- Develop an AIF agent for the SpotMicro system.
- Investigate factors that affect the speed and robustness of learning to walk.
- Reflect on what has been achieved and discuss with BIASlab's researchers.
- Write a report detailing your agent's properties and behaviour.

Supervision

- Weekly progress meetings with dr. Wouter Kouw.
- Periodic progress meetings with BIASlab researchers.
- The student should prepare for meetings in advance, e.g. with notebooks.
- All software that has been developed should be accessible online through BIASlab's github organization and should be legible / usable for future students.

Timeline

The project is available from September 2022 onwards. The total duration will be 32 weeks. Halfway through, there will be a "midterm" evaluation where the student must report on their activities and indicate how they will proceed for the remainder of the project. At the end, the student will write a thesis summarizing their work, their findings and possible future steps. The thesis will be presented in an official "defense" ceremony and a committee of experts will grade the student's work.

3 Parallel reactive computing

Problem description

In this project you are challenged to setup factor graph-based agents that operate simultaneously on multiple CPU cores. You will investigate the feasibility of splitting a factor graph model into multiple parts and how to run message passing-based inference in parallel using multiple CPU cores (see Figure 3). Since graph models can become quite large (e.g., millions of nodes) and data may arrive at different graph locations at different time scales, it is practically important to run these kinds of models in parallel.

In the second part of the project, you will evaluate the performance of parallel message passing-based inference on factor graphs. In principle, parallel inference on multiple CPU's should lead to more efficient use of computational resources and provide the same results in less time.

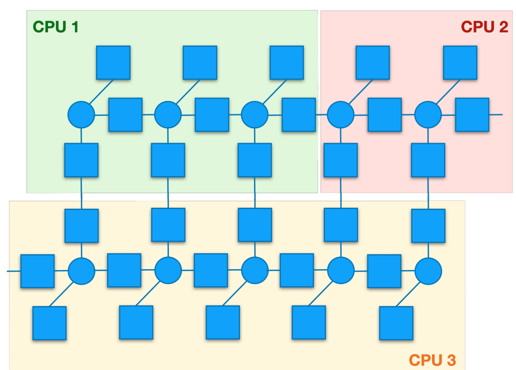

Figure 3: Forney-style factor graph distributed over multiple CPU cores.

Student task description

- Literature search: he student needs to be able to obtain a working understanding of Bayesian Inference, probabilistic methods, probabilistic graphical models, Forney-style factor graphs and message passing.
- Learning programming: this project is best suited for a student with a strong computer science background. The student should get familiar with relevant software tools (Julia programming language, ForneyLab.jl) and understand how to setup parallel computation infrastructures.
- Develop an efficient **parallel implementation** of message passing-based inference in factor graph models.

- Depending on progress, the student may extend the system to more complex benchmark tasks.
- Iterate steps 3 and 4 until satisfaction and/or time runs out.
- Write a MSc-thesis detailing performance benchmarks, results and findings, including positioning of the work within the broader field.

Project positioning

BIASlab (FLUX-7.060) is a subgroup of the Signal Processing Systems (SPS) group that aims to develop **Intelligent Autonomous Agents**. These agents interact with their environment through their sensors and actuators in order to learn purposeful behavior without any supervision ("autonomously"), e.g., to navigate, play soccer or they may learn to decode speech signals under bad acoustic conditions. Our research projects are inspired by the latest insights from machine learning, computational neuroscience and signal processing.

Intelligent agents process information through Bayesian inference. In our lab, we realize these agents as (Forney-style) **Factor Graphs** that execute Bayesian inference in complex models through message passing in a graph. In our team, we are developing a toolbox that supports message passing-based inference in factor graphs, see ForneyLab and ReactiveMP.

Supervision:

- Weekly progress meetings with Dmitry Bagaev.
- The student should prepare update meetings, preferably with derivations or visualizations in interactive notebooks.
- All developed code should be accessible (e.g., on BIASlab's Github organization).

Timeline

The project is available from September 15th 2021 onwards. The total duration will be 32 weeks. Halfway through, there will be a "midterm" evaluation where the student must report on their activities and indicate how they will proceed for the remainder of the project. At the end, the student will write a thesis in 10-page double-column IEEE Transactions style summarizing their work, their findings and possible future steps. The thesis will be presented in an official "defense" ceremony and a committee of experts will grade the student's work.

4 Football player simulation for coach assistance

Problem description

This project is part of a long-term research program that aims to develop an intelligent agent (software) that provides strategic football (soccer) coaching decisions in real-time. We work together with the KNVB (Royal Dutch Football Association), and we aim to use the agent to assist coaching staffs of Dutch football teams.

As an example, consider an agent that receives the real-time coordinates of the players and the ball while the match is ongoing. The agent should be capable of analyzing the match in real-time and make coaching recommendations, e.g., switch our left-wing attacker to the right wing because the pairing with the opposing team's left-back is more advantageous.

Figure 4: Arjan Robben's big chance at the World Championship in 2010.

Intelligent decision making derives from our capacity to compare (the performance of) simulations of future scenarios. In this project, we aim to develop an agent that supports this type of "human intelligence"-inspired decision making. Technically, in this project you will explore the Free Energy Principle (FEP) as a guide on how to develop this type of agents. The FEP is a celebrated neuroscientific theory of how brains perceive and make decisions, based on a single objective, namely "Free Energy" minimization. As such, in a broader context, this project aims to develop a novel neuro-inspired approach to real-time intelligent decision making.

Clearly, developing such an agent is too difficult to accomplish in a single project. Therefore, this project is part of a larger program that encompasses multiple projects, where each project tries to accomplish intermediate milestones.

Student task description

Your main job will be to develop an agent (a software program) that is able to cast strategic coaching decisions during a football game.

Concrete tasks:

- Specify the scope of the project. A first task is to specify what ”strategic coaching decisions” means in this project. We will need to describe and unpack this problem into a concrete and limited problem statement. This stage will be executed through discussions with your coaches and the KNVB.
- Do a literature study on existing approaches to data-driven coaching agents. Do a focused literature review and a select (at least 1) data-driven coaching algorithm from the literature that you will use as your ”reference” algorithm. Implement this reference algorithm.
- Do a literature study on the Free Energy Principle. In this project, our aim is to develop a new approach to data-driven coaching that learns through Bayesian machine learning methods. This can be described as an application of the Free Energy Principle (FEP). You will need to read about the FEP in general and about some previous efforts by BIASlab members and others on how to build FEP-inspired synthetic agents. Together with the literature study on existing coaching agents, this should lead to a small review paper (which we later embed in your thesis).
- Develop a FEP-based agent that learns to cast coaching decisions. The task of designing a new agent based on Bayesian machine learning may seem daunting, but you will be able to design your first agents after a few discussions with your supervisor.
- Evaluate your FEP-based agent in comparison to the reference method(s).
- Iterate steps 4 and 5 until satisfaction and/or time runs out.
- Write and defend an awesome thesis on a new data-driven approach to developing strategic football coaching agents.

Project positioning

The Bayesian Intelligent Autonomous Systems lab (BIASlab) is a subgroup of the Signal Processing Systems (SPS) group that develops agents that interact with their environment through their sensors and actuators. The goal is to learn purposeful behavior, e.g., to navigate, play soccer or they may learn to decode speech signals under bad acoustic conditions. Our research projects are inspired by the latest insights from probabilistic machine learning, computational neuroscience and signal processing.

The KNVB is the Royal Dutch Football Association. You will be working with data scientists from the KNVB who have a tremendous amount of experience analyzing football matches and the behaviour of players. They have collected rich data sets on all matches played by the Dutch national team over the past few years. Their insights will be invaluable for the development of a useful football simulation model.

Supervision:

- Weekly progress meetings with dr. Wouter Kouw and prof. Wim Nuijten.
- Periodic progress meetings with contacts at the KNVB.
- The student should prepare update meetings, preferably with derivations or visualizations in interactive notebooks.

Timeline

The project is available from September 2021 onwards. The total duration will be 32 weeks. Halfway through, there will be a ”midterm” evaluation where the student must report on their activities and indicate how they will proceed for the remainder of the project. At the end, the student will write a thesis summarizing their work, their findings and possible future steps. The thesis will be presented in an official ”defense” ceremony and a committee of experts will grade the student’s work.

5 Active sensing for smart bridges

Problem description

Smart bridges contain sensors that monitor structural integrity. To be precise, the sensors measure the amount of deformation caused by traffic crossing the bridge (see Figure 5). Over time, cracks in the concrete cause a larger deformation for a truck of equal weight. With this data, one can predict more accurately when the bridge needs maintenance: too early means unnecessary costs and too late means dangerous situations.

Figure 5: Stephenson Viaduct Leeuwarden is an example of a Smart Bridge. It contains sensors measuring the deformation caused by traffic crossing the bridge.

Currently, multiple sensors are placed at various locations on the bridge. These are not cheap and installing them costs time. Our first question in this project is: can we position sensors such that we need less of them? Secondly, data collected by these sensors seems redundant and processing costs more than we believe is necessary. That leads to the second question: can we ”silence” individual sensors when their input is not needed? In order to answer these questions, we hope to design an Active Inference agent that listens to the bridge only when necessary.

Student task description

This pilot project tests the usefulness of active inference agents for smart bridges. The student will be tasked with designing an agent that seeks out the most informative sensor data. Its goal will be to either recommend removing a specific sensor because it does not need its input for a subsequent structural integrity model, or to silence one unless some surprising observation is made. ”Silencing” means the sensor is told not to communicate its observation to a central hub. Antea Group will provide a small-scale simulation environment on which experiments can be performed.

Concrete tasks:

- Familiarize yourself with the literature on active inference.
- Obtain a working understanding of the physics of bridges.
- Discuss with researchers at BIASlab and engineers at Antea Group.
- Develop and agent that actively listens to deformation sensors.
- Experiment with the implementation and compare to baselines.
- Analyze results and reflect on what has been achieved.
- Write a report detailing the advantages and limitations of this approach.

Project Positioning

The student will be part of the Bayesian Intelligent Autonomous Systems lab and will collaborate with Antea Group.

The Bayesian Intelligent Autonomous Systems lab (BIASlab) is a subgroup of the Signal Processing Systems (SPS) group that develops agents that interact with their environment through their sensors and actuators. The goal is to learn purposeful behavior, e.g., to navigate, play soccer or they may learn to decode speech signals under bad acoustic conditions. Our research projects are inspired by the latest insights from probabilistic machine learning, computational neuroscience and signal processing.

Antea Group is an engineering consultancy firm that designs civil structures, such as bridges and parks. This collaboration focuses on the Stephenson Viaduct in Leeuwarden, a newly constructed ”smart bridge”. It is projected to last until 2043, but Antea believes it will still be structurally sound long after that. They are currently monitoring its structural integrity to support that claim. The hope is that this will be the first of many smart bridges.

Supervision:

- Weekly progress meetings with Wouter Kouw.
- Periodic progress meetings with contacts at Antea Group.
- The student should prepare meetings, preferably with interactive notebooks.
- All developed code should be accessible online (i.e. Github, Gitlab).

Timeline

The project is available from February 2020 onwards. The total duration will be 32 weeks. Halfway through, there will be a ”midterm” evaluation where the student must report on their activities and indicate how they will proceed for the remainder of the project. At the end, the student will write a thesis in 10-page double-column IEEE Transactions style summarizing their work, their findings and possible future steps. The thesis will be presented in an official ”defense” ceremony and a committee of experts will grade the student’s work.

6 Invertible Bayesian Neural Networks

Problem description

Probabilistic models are a class of machine learning models in which we try to retain uncertainty about the variables inside our model. In these models, which we can represent as factor graphs, we can perform efficient local computations by passing messages through the model. These messages can be thought of as summaries from the part of the graph where they are coming from. This entire approach is based on theories from the field of computational neuroscience. Although this approach is very powerful, it is somewhat limited to the individual factors (operations) that we can use in the probabilistic model.

Standard neural networks are capable of modeling highly non-linear relationships. These networks can be implemented in factor graphs, leading to structures such as the extended Kalman filter. Standard neural networks, however, do not allow for flexible inference as we cannot easily reason back from the output to the input of the network. Therefore we need to guarantee the invertibility in the network. Despite their huge expressive power, these networks are prone to overfitting, when they approximate the supplied data very well, but fail to generalize to similar data sets.

Bayesian neural networks are neural networks which allow for incorporating uncertainty over the intermediate outputs and parameters. However, Bayesian neural networks are not invertible, something which we require for performing efficient inference through this model. Therefore in this project we are interested in exploring model structures, inspired by invertible neural networks, which both allow for efficient inference and simultaneously capture uncertainty over the model parameters

Student task description

- Develop a good understanding of invertible neural networks, Bayesian neural networks, factor graphs and message passing.
- Learn programming in the scientific programming language Julia.
- Explore different training methods for Bayesian neural networks.
- Implement training algorithms for Bayesian neural networks.
- Combine Bayesian neural networks with invertible neural networks and describe how inference can be performed in these models.
- Write a report and present your work.

Project positioning

The Bayesian Intelligent Autonomous Systems lab (BIASlab) is a subgroup of the Signal Processing Systems (SPS) group that develops agents that interact with their environment through their sensors and actuators. The goal is to learn purposeful behavior, e.g., to navigate, play soccer or they may learn to decode speech signals under bad acoustic conditions. Our research projects are inspired by the latest insights from probabilistic machine learning, computational neuroscience and signal processing.

Besides the project activities we highly value a good collaboration with our research group (BIASlab). Besides weekly meetings with the supervisor, you will be involved in weekly update meetings with fellow master and PhD students. Furthermore you will be joining the weekly seminars in which each of us presents our state-of-the-art work.

Timeline

The project is available from September 2021 onwards. The total duration will be 32 weeks. Halfway through, there will be a ”midterm” evaluation where the student must report on their activities and indicate how they will proceed for the remainder of the project. At the end, the student will write a thesis summarizing their work, their findings and possible future steps. The thesis will be presented in an official ”defense” ceremony and a committee of experts will grade the student’s work.

7 Drift control in TwinScan lithography

Problem description

TwinScan lithography systems are used at a very high speed to produce integrated circuits which requires sub-nanometer accuracy and precision. Due to hardware imperfections, these TwinScan systems drift, which causes a negative impact on both the (sub-)nanometer scale positioning and on the system throughput, as it requires (semi-manual) maintenance actions to reduce the impact of these drift.

The current drift controller (see Figure 6) is a static event-based feedback controller which gets activated only when it is necessary by means of a human request or when the system's internal maintenance kicks in. This leads to several challenging issues, e.g.,

1. The measurements are not ideal, i.e., they are subject to measurement noise which are of time-varying and colored nature, unpredictable jumps, and time lags that are introduced while performing measurements.
2. Drifts often possess a different dynamical structure over time for each machine. As a result, using a fixed-model structure is not optimal.

In this project you are challenged to design an improved drift controller based on modern machine learning technology.

Figure 6: TWINSCAN NXE:3350B is one of ASML's nanolithography systems.

Student task description

The student will be challenged to use modern machine learning technology to develop a drift controller. The controller should learn online ("on-the-fly") from past measurements (while continuously providing feedback corrections) so as to cope with the time-varying dynamical nature of the drifts.

We expect that the proposed drift controller has several intrinsic characteristics:

- It should deal with non-uniform measurements in order to update the predictor's (internal) states.
- It should be rather general, in the sense that in principle any type of drift can be handled.
- It should possess a smart decision-making capability by learning from past measurements in order to differentiate between jumps, outliers and handle time-varying noises.
- It should not depend on a fixed-model structure (per class of drifts) as this would violate the previous requirement.

The performance of the proposed controller should be compared to the existing (non-adaptive) event-based drift controller, with both benchmark and real measurement data.

Project positioning

The Bayesian Intelligent Autonomous Systems lab (BIASlab) is a subgroup of the Signal Processing Systems (SPS) group that develops agents that interact with their environment through their sensors and actuators. The goal is to learn purposeful behavior, e.g., to navigate, play soccer or they may learn to decode speech signals under bad acoustic conditions. Our research projects are inspired by the latest insights from probabilistic machine learning, computational neuroscience and signal processing.

ASML is a market and innovation leader in the semiconductor industry. They provide chipmakers with everything they need – hardware, software and services – to mass produce patterns on silicon through lithography. Their machines are incredibly precise, even up to nanometer-scale. ASML employs a large amount of industrial researchers and is quick to incorporate new technologies into their fabrication process.

Supervision:

- Weekly progress meetings with Bert de Vries.
- Periodic progress meetings with Raaja Ganapathy Subramanian at ASML.
- The student should prepare update meetings, preferably with derivations or visualizations in interactive notebooks.

Timeline

The project is available from Augustus 2020 onwards. The total duration will be 32 weeks. Halfway through, there will be a "midterm" evaluation where the student must report on their activities and indicate how they will proceed for the remainder of the project. At the end, the student will write a thesis summarizing their work, their findings and possible future steps. The thesis will be presented in an official ”defense” ceremony and a committee of experts will grade the student’s work.

8 Monitoring health through wearables

Problem description

This is a project in collaboration with Philips on inferring a patient's health condition from wearable sensors. In the TRICA study, patients were sent home after elective surgery with two wearable devices, a patch and a wristband - see pictures, that measured several biometric signals (including heart rate, respiration rate, activity level (acticounts) and activity type (walking, running, cycling, other)). One goal of the TRICA study is to develop algorithms that, based on the measured data streams, detect in real-time when possible post-operative complications surface. This type of analysis is hindered by the fact that the recorded data is often unreliable or missing due to movement and other artifacts.

Figure 7: The TRICA study (which stands for Transitional Care) collects data from wearables for post-operative monitoring of recovery and potential complications.

In this graduation project you are challenged to develop a robust algorithm that, based on the recorded TRICA data base, forecasts heart rate and respiratory rate from activity level and type. In particular, we are interested in personalized algorithms that improve as more data is collected. In other words, your challenge is to develop an algorithm that both executes (predicts HR and RR) and learns to improve its performance as time moves on. The predicted HR and RR can then be used for forecasting health deterioration.

Student task description

We expect the following tasks in this project:

- Specify the scope of the project.
  A first task is to specify the scope of the project. Together with you we will develop the above description into a concrete problem statement that allows you to do a focussed literature review.

- Do a literature study on available methods.
  Do a focussed literature review and a select (at least 1) prediction algorithm from the literature that you will use as your ”reference” algorithm. Implement this reference algorithm.

- Do a literature study on Bayesian machine learning-based prediction.
  At BIASlab, we develop Bayesian prediction algorithms such as advanced Kalman filters that map very nicely to the tasks in this project. You will study the literature in this field and summarize the literature and your plan in a small review paper (which you can later embed in your thesis).

- Develop an online Bayesian machine learning-based prediction algorithm for your task.

- Evaluate your algorithm in comparison to the reference method(s).

- Iterate steps 4 and 5 until satisfaction and/or time runs out.

Project positioning

The project will be carried out at Philips Research and at BIASlab (FLUX-7.060), which is a subgroup of the Signal Processing Systems (SPS) group. BIASlab's research projects are inspired by the latest insights from machine learning, computational neuroscience and signal processing. You will get support (supervision) both from a senior Philips researcher and BIASlab researcher.

Supervision

- Weekly progress meetings with Bert de Vries.

- Periodic progress meetings with Reinder Haakma at Philips.

- The student should prepare update meetings, preferably with derivations or visualizations in interactive notebooks.

Timeline

The project is available from October 2020 onwards. The total duration will be 32 weeks. Halfway through, there will be a ”midterm” evaluation where the student must report on their activities and indicate how they will proceed for the remainder of the project. At the end, the student will write a thesis summarizing their work, their findings and possible future steps. The thesis will be presented in an official ”defense” ceremony and a committee of experts will grade the student's work.

9 Bayesian system identification

Problem description

System identification is concerned with estimating a dynamical model from paired input and output data. In particular, we are considering an electro-mechanical positioning system (EMPS), where a motor drives a small platform on a track between two points (a prismatic joint). These are ubiquitous in robotics and machine tool applications, but can be challenging to control when non-linearities such as friction are involved. Our goal is to estimate the inverse dynamical model (torque/force as a function of position, velocity and acceleration), outperforming the current least-squares baseline approach. More information on the electro-mechanical positioning system, including baseline, data and code, can be found at `http://nonlinearbenchmark.org/`.

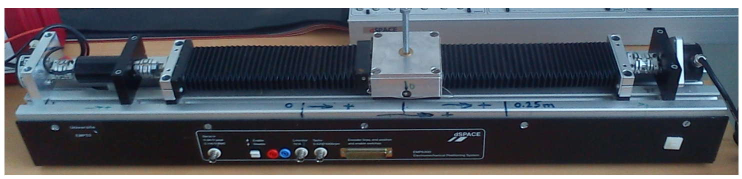

Figure 8: Electro-mechanical positioning system setup.

Student task description

We expect the student to familiarize him/her self with variational free energy minimization and system identification. Your job will be to program an agent to infer the dynamical parameters of the system from given input-output data, and compare the agent to baseline methods. We hope to include an *optimal design / active inference* experiment as well, where the agent infers which inputs it should test to infer the dynamics as fast as possible.

Concrete tasks

:

- Literature search over the intersection between Bayesian machine learning and nonlinear system identification.
- Familiarize yourself with the difficulties of nonlinear system identification.
- Understand the challenges of applying variational free energy minimization to nonlinear system identification.
- Collaborate and discuss with researchers in BIASlab and the Control Systems group.

- Implement an agent that infers dynamics in the system.
- Experiment with the implementation and compare to baselines.
- Analyze results and reflect on what has been achieved.
- Write a report detailing the advantages and limitations of this approach.

Project positioning

The Bayesian Intelligent Autonomous Systems lab (BIASlab) is working on intelligent agents that perceive and act through Bayesian machine learning. We work with a principled technique for approximate inference, called variational free energy minimization, and apply it to discrete-time dynamical systems. For an introduction to this framework, see our toolboxes ForneyLab and ReactiveMP. Currently, we are interested in system identification and would like to tackle the EMPS benchmark using our techniques.

Supervision

:

- Weekly progress meetings with Maarten Schoukens and Wouter Kouw.
- The student is expected to be prepared for meetings, preferably by writing interactive notebooks.
- All developed code and reports should be accessible online to achieve efficient collaboration.

Timeline

The project is available from August 2022 onwards. The total duration will be 32 weeks. Halfway through, there will be a ”midterm” evaluation where the student must report on their activities and indicate how they will proceed for the remainder of the project. At the end, the student will write a thesis summarizing their work, their findings and possible future steps. The thesis will be presented in an official ”defense” ceremony and a committee of experts will grade the student’s work.

10 Graphical editor for building agents

Problem description

You are challenged to develop a graphical web-based editor/debugger for developing and monitoring the dynamics of simulated agents. Please have a look at this video by Bret Victor that demonstrates the essence of the envisioned editor. Rather than developing electronic circuits, in this project we focus on developing intelligent agents that minimize a quantity known as Free Energy through message passing in a factor graph. See the 5SSD0 Bayesian Machine Learning & Information Processing lesson on intelligent agents for more information.

In the second part of the thesis work, you use the editor/debugger to analyze the internal dynamics (i.e., monitor the free energy distribution) of a few well-known (but not well-understood) intelligent agents, thus leading to an increased understanding of how Free Energy minimizing dynamics give rise to intelligent behavior (see e.g. this video for a neuroscience perspective.)

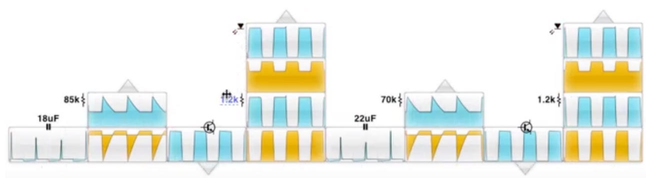

Figure 9: Visualization of (electronic circuit) dynamics (source.

Student task description

Please note that, aside from having a demonstrated interest in dynamic systems modeling and/or machine learning, for this project it is essential that you also have good programming skills, in particular related to web front-end technologies. The project is ideally suited to a good programmer with interests in visual editing and machine learning. It is also possible to do this project with 2 people.

The task is organized into the following steps:

- Scope: a first task is to narrow the scope of this project. Development of a good editor/debugger may take years, so we will focus on a few simple features that showcase proof-of-concept and allow dynamic monitoring of free energy distribution during simulations.
- Literature study of modern supporting technology for development of web-based editors.
- Literature study of free energy minimization of dynamic systems in factor graphs, finalized by an interim report and project planning.

- Develop/update a simple web-based graphical editor/debugger based on Bret Victor's principles.
- Analyze a highly performant but not well-understood intelligent agents by online monitoring and manipulation of the internal dynamics of these agents.
- Iterate steps 4 and 5 until satisfaction and/or time runs out.
- Write and defend an awesome thesis on a new approach to interpretable AI through online visualization and manipulation of the simulated system.

Project positioning

The Bayesian Intelligent Autonomous Systems lab (BIASlab) is a subgroup of the Signal Processing Systems (SPS) group that develops agents that interact with their environment through their sensors and actuators. The goal is to learn purposeful behavior, e.g., to navigate, play soccer or they may learn to decode speech signals under bad acoustic conditions. Our research projects are inspired by the latest insights from probabilistic machine learning, computational neuroscience and signal processing.

Supervision:

- Weekly progress meetings with Bert de Vries.
- The student should prepare update meetings, preferably with derivations or visualizations in interactive notebooks.
- All developed code should be accessible (e.g., on BIASlab's Github organization).

Timing

The project is available from April 26th 2020 onwards. The total duration will be 32 weeks. Halfway through, there will be a "midterm" evaluation where the student must report on their activities and indicate how they will proceed for the remainder of the project. At the end, the student will write a thesis summarizing their work, their findings and possible future steps. The thesis will be presented in an official "defense" ceremony and a committee of experts will grade the student's work.