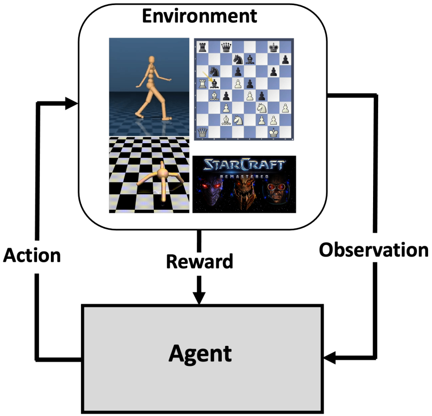
Reinforcement Learning-based Artificial Pancreas Systems to Automate Treatment in Type 1 Diabetes

Chirath Hettiarachchi

Research Fellow,
School of Computing, College of Engineering, Computing & Cybernetics
Australian National University

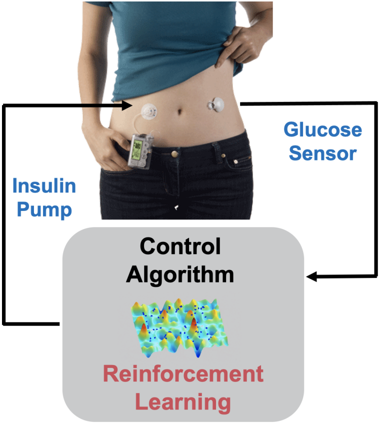
July 1, 2024

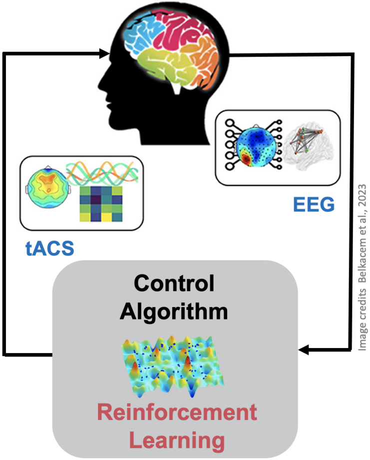
Background: Reinforcement Learning (RL)



The "Law of Effect" in psychology introduced learning by trial and error, which described the effect of reinforcing events based on the tendency to select actions [Thorndike, 1911].

Reinforcement Learning for Health (RL4H)

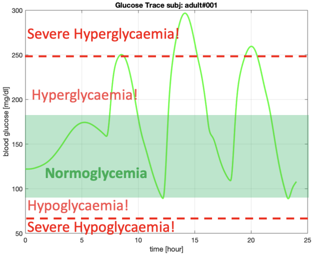




The Glucoregulatory System

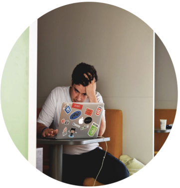
Sleep

Meals



Exercise

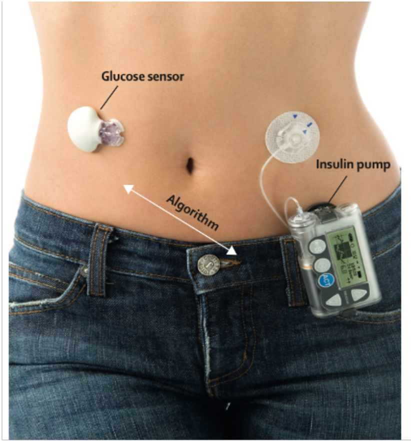
Stress



Maintaining glucose homeostasis is vital!

Clinical objective is to improve Time in Normoglycemic Range (TIR), while avoiding hypoglycemic and hyperglycemic risk.

Artificial Pancreas Systems (APS)



APS are **high-risk medical devices**. Existing commercial APS are **hybrid systems (manual decision-making required)**, and designed using classical control algorithms (PID & MPC).

Image Credits: Atkinson et al 2014 , Medgadet.

Existing Basal-Bolus Clinical Treatment Strategy

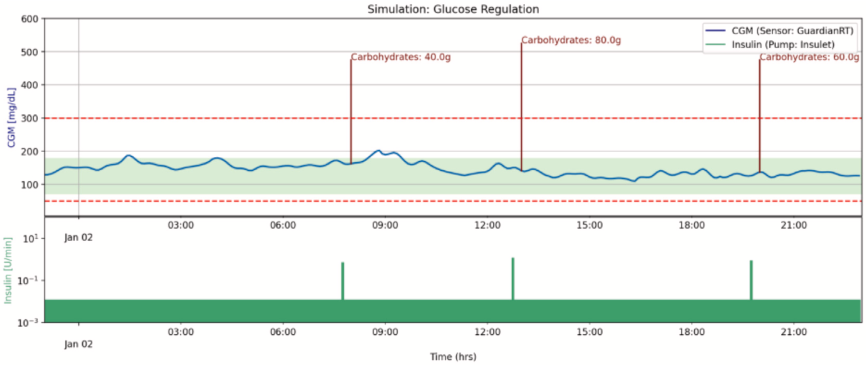
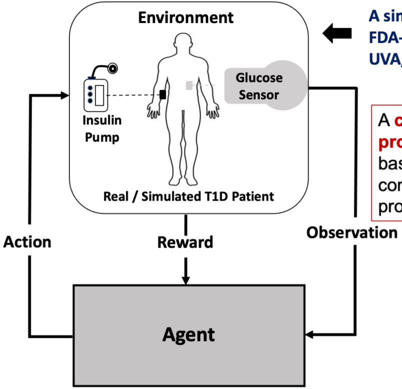


Figure: A basal-bolus insulin treatment strategy.

Requires manual user input on meal announcements & carbohydrate estimation, which leads to erros and sub-optimal glucose regulation.

Method: Problem Formulation

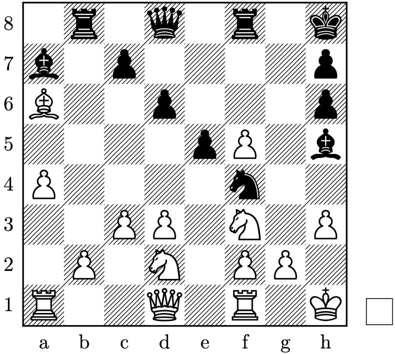


A simulator based on the FDA-approved UVA/Padova 2008 Model.

A **challenging meal protocol** (randomised based on CHO content, time, & probability.

Hettiarachchi, Chirath, et al. "A Reinforcement Learning Based System for Blood Glucose Control without Carbohydrate Estimation In Type 1 Diabetes: In Silico Validation", EMBC2022.

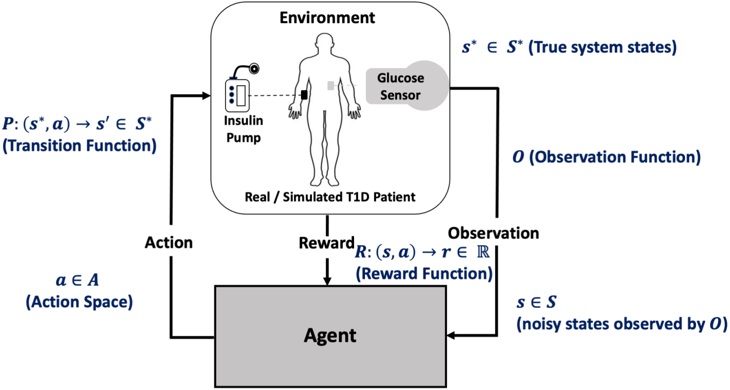
Method: Challenges in Real-World RL (RWRL)



Stockfish 8 vs AlphaZero
London 2018(12) (0-1)

RWRL Challenges: Complex & only partially observable, uni-directional control, safety, explainability, large delays, uncertsainty & disturbances, unspecified reward functions...

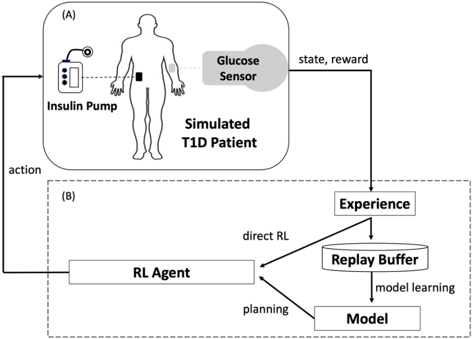
Method: Problem Formulation



The problem was formulated as a Partially Observable Markov Decision Process (POMDP).

Method: G2P2C Algorithm

- G2P2C consists of 3 sequential updates:
  1. PPO update,
  2. **Auxiliary model learning** update,
  3. **Short-term planning** update.



Hettiarachchi, Chirath, et al. "G2P2C—A modular reinforcement learning algorithm for glucose control by glucose prediction and planning in Type 1 Diabetes." Biomedical Signal Processing and Control(2024).

Method: Experiment Setup & Benchmark Algorithms

Experiment Setup: Following established best practice **Simglucose** (Xie, 2018), an open-source T1D simulator based on FDA-approved UVA/PADOVA 2008 model was used with a challenging meal protocol.

Clinical Algorithms

- Two standard treatment approaches were replicated.
  1. Basal Bolus Ideal (**BBI**) - **perfect meal information**.
  2. Basal Bolus **Human Error** (**BBHE**).
- Both methods used **meal announcements 20-minutes in advance**.

RL Algorithms

- A2C (Advantage Actor Critic).
- PPO (Proximal Policy Optimisation).
- SAC (Soft Actor Critic).
- G2P2C (model learning + planning).

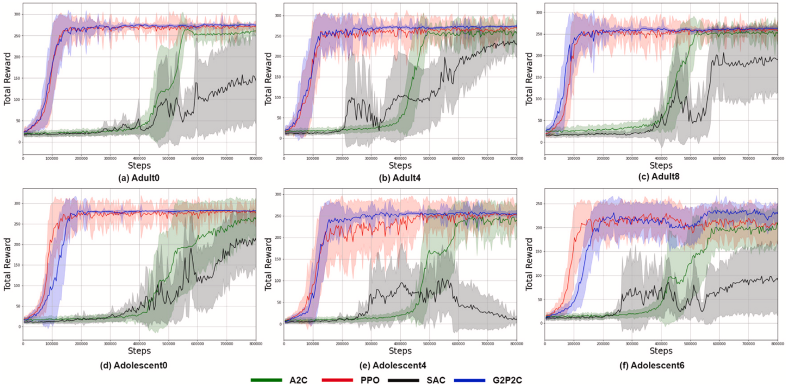
Results: Clinical Analysis

- G2P2C achieved a higher TIR of 72.69% for the adult cohort compared to BBI (71.02%) and BBHE (69.78%), while outperforming all RL algorithms.

Algorithm	TIR	Failures	Total reward
Adults			
BBI	71.02 ± 11.29	0.39	–
BBHE	69.78 ± 11.29	0.35	–
A2C	59.06 ± 14.31	9.11	244 ± 47
PPO	69.12 ± 10.53	2.79	264 ± 26
SAC	61.76 ± 21.01	59.49	146 ± 98
G2P2C	72.69 ± 9.53	1.62	268 ± 21
Adolescents			
BBI	71.43 ± 12.31	0.00	–
BBHE	70.23 ± 12.52	0.00	–
A2C	56.03 ± 14.40	14.41	227 ± 48
PPO	63.72 ± 13.95	4.93	249 ± 31
SAC	65.62 ± 20.16	82.06	107 ± 89
G2P2C	64.33 ± 13.18	1.48	254 ± 22

Hettiarachchi, Chirath, et al. "G2P2C—A modular reinforcement learning algorithm for glucose control by glucose prediction and planning in Type 1 Diabetes." Biomedical Signal Processing and Control(2024).

- A2C & SAC performed the lowest. SAC had the highest failures. PPO & G2P2C were the best RL algorithms.
- G2P2C improved safety, reducing catastrophic failures to 1.62% & 1.48% in adults & adolescents compared to 2.79% & 4.93% in PPO.



Results: RL Treatment Strategy (G2P2C)

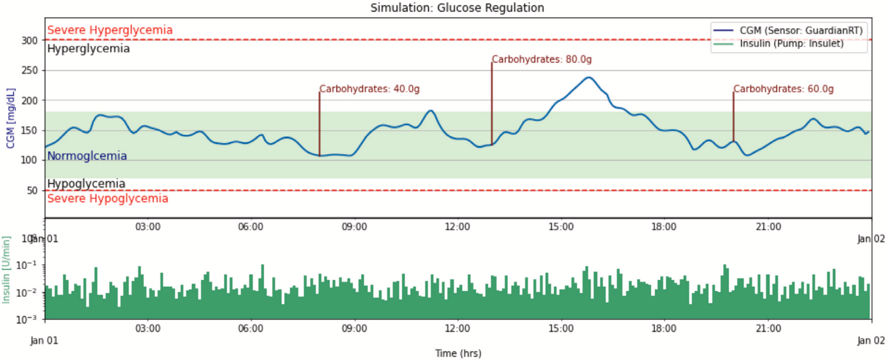
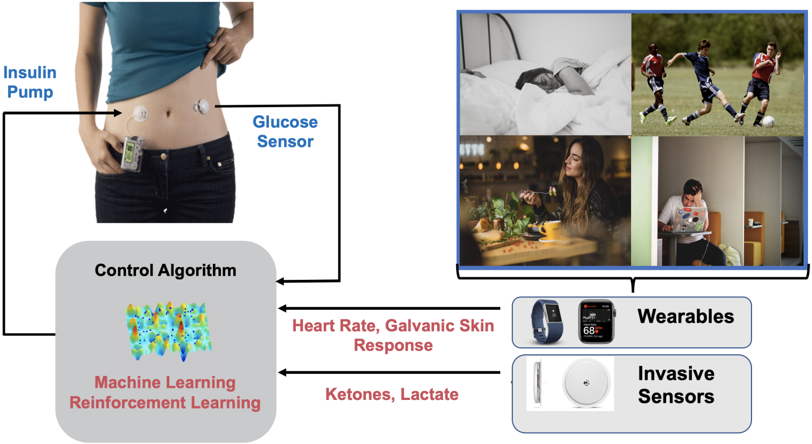


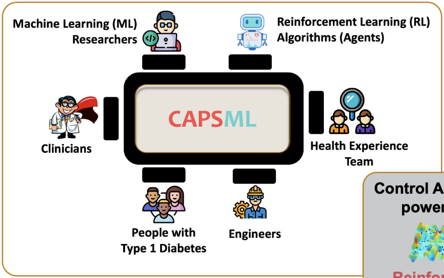
Figure: A RL-based treatment strategy.

No meal announcements.
No meal carbohydrate (CHO) estimation.

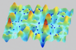
Conclusion & Future Work



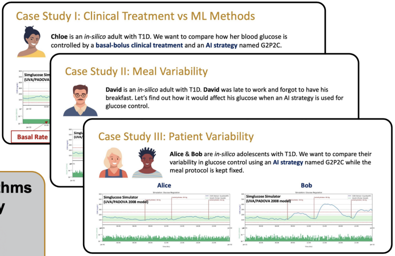
Resources: CAPSML (*capsml.com*)



Control Algorithms powered by



Reinforcement Learning

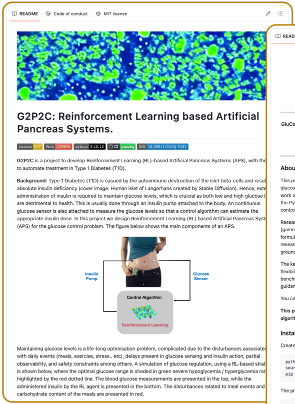


Alice
Age: 18, Gender: F, Weight: 68.7Kg
Total Daily Insulin (TDI): 36.73
Insulin to Carbohydrate Ratio (ICR): 12
Insulin Sensitivity Factor (ISF): 15.03

Select *in-silico* subjects

Learn about your meals
How much carbs are in a **McDonalds Cheese Burger**?
How does it affect glucose?

Resources: GluCoEnv & G2P2C Codebase



README Code of conduct MIT license

G2P2C: Reinforcement Learning based Artificial Pancreas Systems.

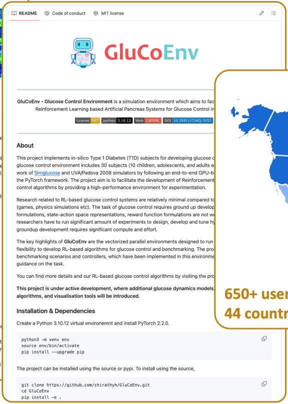
License MIT | Web CAPSML | python 3.10.12 | CI passing | DOI 10.25911/CXAQ-3151

G2P2C is a project to develop Reinforcement Learning (RL)-based Artificial Pancreas Systems (APS), with the
to automate treatment in Type 1 Diabetes (T1D).

Background: Type 1 Diabetes (T1D) is caused by the autoimmune destruction of the islet beta-cells and resul
absolute insulin deficiency (cover image: Human islet of Langerhans created by Stable Diffusion). Hence, exte
administration of insulin is required to maintain glucose levels, which is crucial as both low and high glucose l
are detrimental to health. This is usually done through an insulin pump attached to the body. An continuous
glucose sensor is also attached to measure the glucose levels so that a control algorithm can estimate the
appropriate insulin dose. In this project we design Reinforcement Learning (RL) based Artificial Pancreas Syst
(APS) for the glucose control problem. The figure below shows the main components of an APS.

Maintaining glucose levels is a life-long optimisation problem, complicated due to the disturbances associated
with daily events (meals, exercise, stress.. etc), delays present in glucose sensing and insulin action, partial
observability, and safety constraints among others. A simulation of glucose regulation, using a RL-based strat
is shown below, where the optimal glucose range is shaded in green severe hypoglycemia / hyperglycemia ran
highlighted by the red dotted line. The blood glucose measurements are presented in the top, while the
administered insulin by the RL agent is presented in the bottom. The disturbances related to meal events and
carbohydrate content of the meals are presented in red.

https://github.com/RL4H/G2P2C



README Code of conduct MIT license

GluCoEnv - Glucose Control Environment is a simulation environment which aims to fac
Reinforcement Learning based Artificial Pancreas Systems for Glucose Control in

License MIT | python 3.10.12 | Web CAPSML | DOI 10.25911/CXAQ-3151

About

This project implements in-silico Type 1 Diabetes (T1D) subjects for developing glucose c
glucose control environment includes 30 subjects (10 children, adolescents, and adults e
work of Simglucose and UVA/Padova 2008 simulators by following an end-to-end GPU-b
the PyTorch framework. The project aim is to facilitate the development of Reinforcement
control algorithms by providing a high-performance environment for experimentation.

Research related to RL-based glucose control systems are relatively minimal compared to
(games, physics simulations etc). The task of glucose control requires ground up develop
formulations, state-action space representations, reward function formulations are not we
researchers have to run significant amount of experiments to design, develop and tune hy
groundup development requires significant compute and effort.

The key highlights of **GluCoEnv** are the vectorized parallel environments designed to run
flexibility to develop RL-based algorithms for glucose control and benchmarking. The pro
benchmarking scenarios and controllers, which have been implemented in this environme
guidance on the task.

You can find more details and our RL-based glucose control algorithms by visiting the pro

**This project is under active development, where additional glucose dynamics models,
algorithms, and visualisation tools will be introduced.**

Installation & Dependencies

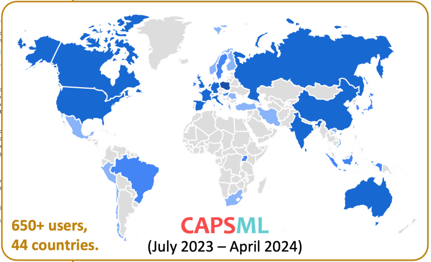
Create a Python 3.10.12 virtual environemnt and install PyTorch 2.2.0.

```
python3 -m venv env
source env/bin/activate
pip install --upgrade pip
```

The project can be installed using the source or pypi. To install using the source,

```
git clone https://github.com/chirathyh/GluCoEnv.git
cd GluCoEnv
pip install -e .
```

https://github.com/RL4H/GluCoEnv



Thank You

Acknowledgement

- This research was funded by the Australian National University (ANU), School of Computing; Our Health in Our Hands (OHIOH) strategic initiative; National Computation Infrastructure (NCI Australia) under the ANU Merit Allocation Scheme.
- **Collaborators**: Dr David O'Neal, Dr Barbora Paldus, & Dr Dale Morrison from the Diabetes Technology Research Group, St Vincent's Hospital.

Dr Chirath Hettiarachchi

Dr Nicolo Malagutti

Prof. Christopher Nolan

Dr Elena Daskalaki

Prof. Hanna Suominen

Australian National University

OHIOH

NCI AUSTRALIA

References I

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. (2016). Openai gym. *arXiv Eprint arXiv:1606.01540*.

Kovatchev, B. P., Clarke, W. L., Breton, M., Brayman, K., and McCall, A. (2005). Quantifying temporal glucose variability in diabetes via continuous glucose monitoring: mathematical methods and clinical application. *Diabetes Technology & Therapeutics*, 7(6):849–862.

Tassa, Y., Doron, Y., Muldal, A., Erez, T., et al. (2018). Deepmind control suite. *arXiv preprint arXiv:1801.00690*.

Todorov, E., Erez, T., and Tassa, Y. (2012). Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE.

Xie, J. (2018). Simglucose v0. 2.1 (2018). *Available at: https://github.com/jxx123/simglucose*.