Table of Contents

1 Introduction

1.1 Objectives

The goal of this project is to progress the development and definition of the PCT-based approach to learning. Ultimately it would be good to get to a generic process that could be applied to, and learn, any control problem.

1.2 Theory

Within PCT, learning concerns memory, problem-solving and reorganisation. Although the latter will be the initial focus of this project, hopefully it will lead to insights into memory and problem-solving.

A few points about how learning and adaptation, of increasing degrees of sophistication, are regarded from a PC perspective:

1. Control - a perceptual control system is inherently adaptive, in that it dynamically varies its output in order to maintain its perceptual input at a desired level. In this way it protects the perceptual input from (unknown) disturbances in the environment. This type of adaptation would not be classified as learning as the parameters of the system do not change; in contrast to the remainder of this list.

2. Auto-tuning - the quality of performance of a pre-organised perceptual control system can be improved by the adjustment (optimisation) of the effect of the error on the output; in other words, the output gain (a system parameter). This can be automatically adjusted over time according to the persistent (long term) error within the system. What is being controlled, the

perception, does not change, but the responsiveness does; for the purpose of increasing performance.

3. Perceptions - the perceptual input, the variable to be controlled, can also potentially be derived through a learning process. A perceptual input function, at any point in the hierarchy of perceptual control systems is generally a weighted sum of perceptions, from the previous, lower level. This hierarchy of perceptual input nodes is similar to a deep learning network. However, the main difference is that the rationale is to reduce the overall error (improve control) in the system rather that of classification. So, the learning process can be driven by that control error rather than a back propagation error based upon classification.
4. Memory - a further level of learning concerns the values of the reference signals (goals) for the perceptual inputs. Although the structure of the hierarchy may be the same, different goal values may be required in different situations. So, the links between nodes may represent not the value of the goals but keys to values of the goals; in other words an associative memory system. The acquisition of these values can be achieved by the system automatically recording them when control is successful during one execution of the game. In a subsequent execution the value can be replayed in order to more swiftly achieve successful control.

Essentially reorganisation (points 2 and 3 above) concerns the change of structure of a perceptual control network by way of altering the organisational parameters of the system; usually meaning connection weights.

The main rationale for reorganisation is to reduce “intrinsic” error in the system. To this end reorganisation only occurs when there is error present. Therefore, structure persists if there is no error, otherwise the structure is reorganised out of existence.

1.3 Models

Aside from conceptual descriptions, formal definition and modelling of reorganisation, within PCT, is fairly limited, but there are some nice demonstrations associated with LCS3 (www.billpct.org), namely **ThreeSys** and **ArmControlReorg**. Both deal with reorganisation of weights on output connections that result in the improvement of quality of control.

1.4 Comparison

In order to identify the benefits, if any, of the PCT approach it will be useful to clarify the differences with other methods and mechanisms by implementing comparative models. Other proposed approaches, which may or may not be complementary, are Reinforcement Learning, Predictive Coding Theory and Hebbian/anti-Hebbian learning.

2 Reorganisation mechanism

As reorganisation will be the main initial focus of this project, listed here are some steps to progress the investigation of the mechanism:

- **Correction adjustment** - revisit and clarify the parameter correction mechanism in terms of connection weights, learning rates, moving or box-car error averaging, variables limits and parameter generalisation. This can be done by modelling reorganisation of a single parameter, gain, in an output function, within the context of parameters of different sizes and different types of disturbances. Although the main learning type to be considered will probably be **e-coli**, it may be useful to consider others.
- **Two parameters –** extend the mechanism with the reorganisation of two parameters, both gain and slowing factor, on a leaky integrator output function.
- **Perceptual gain** – apply to the same parameters on a perceptual function with one input. Would this converge to a value of gain = 1?
- **Perceptual function** – apply the mechanism to the reorganisation of a multi-input perceptual function. The **ThreeSys** model could be adapted for this purpose. Currently it only reorganises output weights.
- **Reorganisation loop** – implement the reorganisation as a control loop that adjusts parameters on the perceptual control loop as a means to reducing its own perception of the absolute (intrinsic) error.
- **Hierarchy –** extend the reorganisation mechanism to a hierarchy of control systems.

3 Applications

Some envisaged applications of control problems to which to apply what is learned from this development are:

- Adaline learning algorithm (https://www.youtube.com/watch?v=3993kRqejHc)
- Mountain car problem – a reasonably simple continuous variable control problem which has been implemented in both reinforcement learning and predictive coding theory, so will act as a useful comparison.
- Tic-tac-toe – A discrete variable, problem-solving task. If this can be implemented as goal selection, perceptual control-based system rather than a state-action pairings this would give great insight into PCT problem-solving learning and program-level control. Again existing implementations by reinforcement learning would be a useful comparison.
- DeepLoco – The **DeepLoco** (http://www.cs.ubc.ca/~van/papers/2017-TOG-deepLoco/) system has produced some very impressive looking results for the control of a humanoid robot simulation. Achieving similar results with a PCT-based system would represent substantial progress and significant validation of the PCT paradigm.

- Images – natural images are a potential “real” environment for a PCT-based learning system. There are parallels between convolutions in visual deep learning and PCT perceptual functions, giving the equivalent of “feature detectors”.

4 Open questions

Some questions to be addressed along the way:

- Is intrinsic error necessary for reorganisation?
- Can intrinsic error it be simulated?
- Can an equivalent of intrinsic error be used?
- Can the error within a sub-hierarchy be used to drive reorganisation within that hierarchy
- Learning without reorganisation – is learning a new task immediately done, without reorganisation?
- How modular is reorganization?
- Can a derivative of a variable (e.g. speed) be learned from the input of that variable?
- Is the concept of reinforcement, with respect to behaviour and learning, valid?

5 Collaboration

As demonstrated by our glorious and most-beloved statesmen of the modern era, Trump and Farage, we can make more progress and are all better off by working in unity together. Therefore, if this is a collaborative project then we can advance PCT far more effectively then working individually. So, if you’d like to contribute in one of the ways listed below, or another way, just jump in:

- Modelling **-** design, test and/or run models of PCT reorganisation with PCT Framework software.
- Provide a brief summary of how memory fits in with PCT learning and reorganisation.
- Provide a brief summary of the PCT approach to problem-solving.
- Describe the reinforcement learning methodology in comparison with PCT learning.
- Describe predictive coding theory in comparison with PCT learning.
- Summarise the methodology of the DeepLoco project. Identify its limitations and how PCT might provide a better, simpler or more efficient solution to the problem.
- Provide a description of why the concept of “reinforcement”, with respect to behaviour, and, therefore, learning is invalid.

Appendix

Notes:

Equivalent representations of the leaky integrator function:

g = gain, s = slowing factor, e = error, o = output, r = g/s = gain rate, d = 1/s = leak rate

1. o = o + (ge-o)/s
2. o = (o+re) - od
3. o = od + ge(1-d) aka exponential smoothing

Reorganisation loop

From [Martin Taylor 2017.12.05.12.42]

possible functional diagram for a "reorganization control loop"