Modulation of temporal decision-making in a deep reinforcement learning agent under the dual-task paradigm

Amrapali Pednekar
Department of Information Technology
IDLab, Ghent University - imec
`Amrapali.Pednekar@UGent.be`

Álvaro Garrido-Pérez
Department of Information Technology
IDLab, Ghent University - imec
`Alvaro.GarridoPerez@UGent.be`

Yara Khaluf
Department of Social Sciences
Wageningen University and Research
`yara.khaluf@wur.nl`

Pieter Simoens
Department of Information Technology
IDLab, Ghent University - imec
`Pieter.Simoens@UGent.be`

Abstract

This study explores the interference in temporal processing within a dual-task paradigm from an artificial intelligence (AI) perspective. In this context, the dual-task setup is implemented as a simplified version of the Overcooked environment with two variations, single task (T) and dual task (T+N). Both variations involve an embedded time production task, but the dual task (T+N) additionally involves a concurrent number comparison task. Two deep reinforcement learning (DRL) agents were separately trained for each of these tasks. These agents exhibited emergent behavior consistent with human timing research. Specifically, the dual task (T+N) agent exhibited significant overproduction of time relative to its single task (T) counterpart. This result was consistent across four target durations. Preliminary analysis of neural dynamics in the agents' LSTM layers did not reveal any clear evidence of a dedicated or intrinsic timer. Hence, further investigation is needed to better understand the underlying time-keeping mechanisms of the agents and to provide insights into the observed behavioral patterns. This study is a small step towards exploring parallels between emergent DRL behavior and behavior observed in biological systems in order to facilitate a better understanding of both.

1 Introduction

How do humans track time? This question has captivated researchers in psychology, neuroscience, and cognitive science for many years [13, 3, 7, 20, 2]. Despite extensive research, there is a fragmented and sometimes conflicting understanding of temporal processing and its underlying mechanisms. However, parallel research across different fields has led to convergence on certain aspects of timing [19, 11]. This reflects both the inherent complexity of the problem and the importance of incorporating complementary tools and perspectives to further deepen our understanding of human timing.

The current study attempts to touch on this problem using an artificial intelligence (AI) perspective. To do so, we consider a well-established and robust finding from timing research. Namely, the modulation of time perception in a dual-task paradigm. Numerous behavioral studies have shown that timing performance is affected by the presence of a concurrent cognitive task [6, 4, 5, 12, 21]. Such findings have implicated shared resources between temporal and cognitive processes. Furthermore,

functional brain imaging studies have shown that temporal processing is distributed across multiple brain regions, and that neurons involved in timing also participate in other cognitive functions [20, 7]. Thus, time is affected by concurrent cognitive tasks possibly because temporal processing is encoded in the activity of neurons that are also involved in other cognitive processes.

We show that deep reinforcement learning (DRL) agents exhibit behavior that suggests temporal interference in a dual-task paradigm. Specifically, two DRL agents were trained separately on a single task and a dual task. The single task variation involved an embedded time production task (referred as single task (T)), while the dual task variation involved the time production task and a concurrent number comparison task (referred as dual task (T+N)). All other task parameters and agent characteristics were kept the same across the two tasks. To substantiate the findings, training was performed separately for four target durations (7, 8, 9, and 10 time steps), resulting in four independently trained agents per task type. Analysis of the distribution of durations produced by the different agents revealed that dual task (T+N) DRL agents significantly overproduced across all durations as compared to their single task (T) counterparts. Thus, the emergent behavior observed in DRL agents under a dual-task paradigm parallels findings from behavioral studies on time production with interference [12, 4, 5].

A preliminary analysis of neural dynamics in both types of agents did not show strong evidence for a dedicated or an intrinsic timer [15]. For some target durations, oscillations with frequencies equal to the target duration were observed in the latent space of neural activities. However, further analysis is needed to confirm that these oscillations correspond to a time-keeping mechanism. Thus, no conclusive neural evidence was found to explain the observed behavioral differences.

2 Methodology
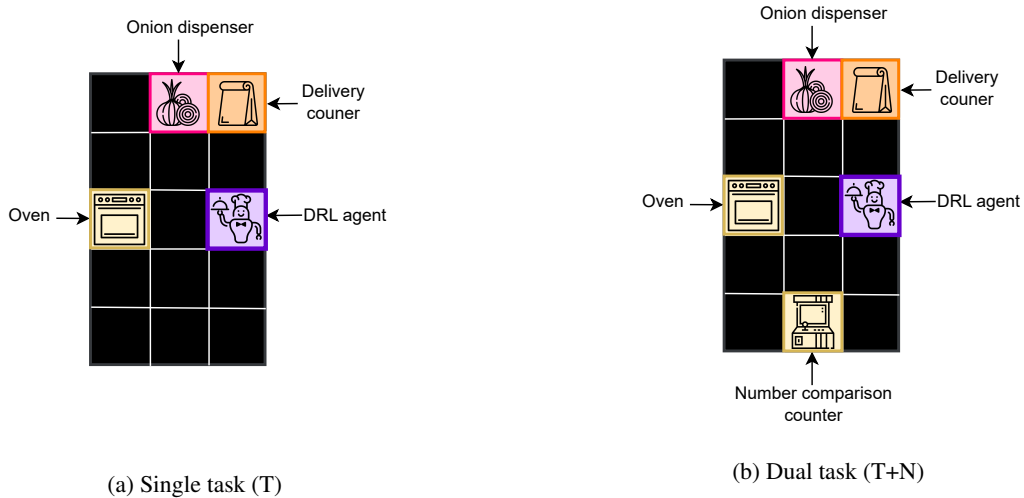

(a) Single task (T)

(b) Dual task (T+N)

Figure 1: Grid worlds representing a simplified version of the OverCooked environment [8], used for the single task (T) and dual task (T+N) experiments (icons were sourced from Flaticon.com).

The Reinforcement learning (RL) environment was a simplified version of the popular OverCooked environment [8]. The modified setup (shown in Figure 1a) consisted of a single-agent scenario with a 5x3 grid world featuring three counters (an onion dispenser, an oven and a delivery counter) and two objects (onion and soup). The agent could perform six actions, five for navigation ('wait', 'up', 'down', 'left' and 'right') and one 'interact' action to engage with an adjacent counter or object.

The agent's goal was to deliver soup to the delivery counter. To do this, it had to pick up an onion from the dispenser, place it in the oven, and wait for the soup to cook. The oven started an internal (invisible) timer upon receiving the onion, tied to a target duration. The soup became ready only after this duration and thus interacting with the oven before that had no effect. Once ready, the agent could take the soup at any point (i.e., at or after the target duration) and deliver it to complete a trial. Each soup delivery yielded a '+1' reward, while all other actions yielded zero reward, even if the soup

stayed in the oven beyond the target time. The agent had to deliver as many soups as possible per episode (consisting of 100 time steps), making accurate timing essential for optimal performance.

In the dual-task variation, a number comparison counter was added to the above setup (shown in Figure 1b). After placing the onion in the oven, this counter activated for 4 time steps (irrespective of the target duration), displaying numbers between 1 and 10. The agent had to respond with the 'interact' action if the number was less than 5, or the 'wait' action if it was 5 or more. It received an immediate '+1' reward for correctly performing the number comparison task and a zero reward otherwise. After four time steps, the number comparison ended, and the task proceeded as in the single task (T) setup. It is important to note that, the agent always had enough time in all target intervals to retrieve the soup at least one time step before the target duration.

The DRL agents were implemented using the recurrent policy variant of the Proximal Policy Optimization (PPO) algorithm from the Stable-Baselines3 (SB3) library [22] (see Appendix). An entropy coefficient of 0.05 encouraged exploration, causing the agent to move around the grid while the soup was cooking. This proved especially helpful in successfully training the dual-task variation. For both tasks, the corresponding agents were trained for 100,000 time steps, which was sufficient to learn the tasks and achieve comparable performance.
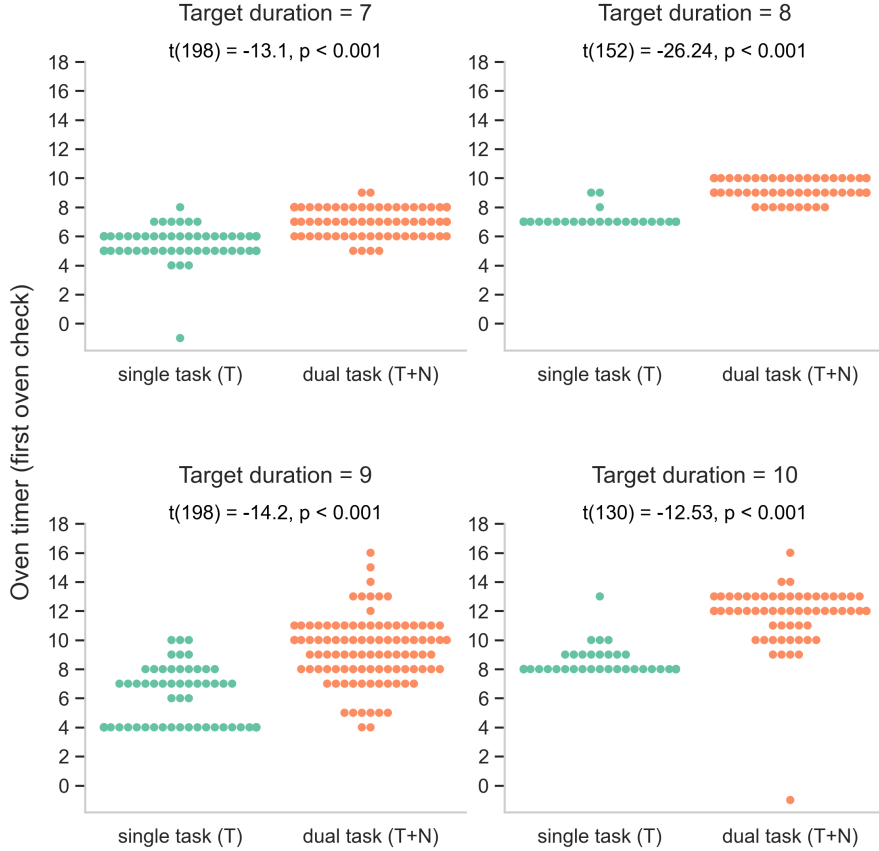

Figure 2: Distribution of oven timers corresponding to the 'first oven check' in 25 episodes (each of 100 time steps) across the two task types, shown for different target durations. The dual task (T+N) agent tends to significantly overestimate ($p < 0.001$) as compared to its single task (T) counterpart. The corresponding independent t-test statistics are shown in the plots.

3   Behavioral analysis

For the behavioral analysis, we define the 'first oven check' as the oven timer value at which the agent initiates the 'Interact' action with an onion carrying oven. If the oven timer is more than or equal to the target duration, agent can take the soup out and the corresponding oven time is recorded

as the 'first oven check' for that trial. In contrast, if the oven timer is less than the target duration, the agent cannot take the soup out. In this case, if the agent consecutively continues to 'Interact' with the oven until it can take the soup out (i.e., until the target duration), the oven timer corresponding to the first 'Interact' action is recorded as the the 'first oven check'. However, if at least one of the consecutive actions is not 'Interact' the oven timer is not recorded. The intuition is that the agent checks the oven because it considers the target duration to be reached and continues to check it until it can get the soup out. Thus, by comparing the 'first oven checks' for both tasks, we can assess whether the concurrent number comparison game induced a change in timing behavior.

Figure 2 shows the distribution of 'first oven checks' across 25 episodes (each consisting of 100 time steps) for the different target durations. It can be seen that the average 'first oven check' is significantly higher ($p<0.001$) in the dual task (T + N) as compared to the single task (T). The independent t-test statistics and degrees of freedom are shown in Figure 2. Thus, on average, the dual task (T+N) agents exhibit a significant overproduction of time relative to their single task (T) counterparts.

It is important to note that in the dual tasks, the agents had enough time after the number game stopped to go to the oven and 'Interact' with it at least one time step before the target duration. Moreover, despite being trained for the same number of training steps, the dual task (T+N) agents achieve performance comparable to their single-task (T) counterparts. Figure 3 shows the average number of soups produced across the 25 episodes in the two task types, for different target durations. Since an agent can only produce a new soup after delivering the previous one, Figure 3 reflects agent performance. Even in the least favorable case (target duration 10), the dual task (T+N) agent produces ~53% of the soups produced by its single task (T) counterpart.
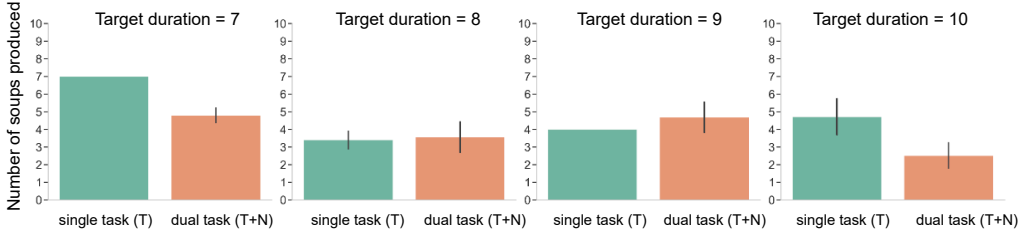

Figure 3: Average number of soups produced across the 25 episodes (each of 100 time steps) for the two task types, shown for different target durations.

Thus, behaviorally, the DRL agents seem to replicate the well-established findings in human timing, namely, overproduction of time in the presence of a concurrent cognitive task [12, 4, 5]. It is important to note that this was an emergent behavior in the DRL agents. The agents were not explicitly provided with a timer or biologically inspired neural structures to influence their behavior in order to be closer to biological systems. While previous studies have qualitatively reported emergent timing biases in deep neural networks (DNNs) [10, 23], to the best of our knowledge, this is the first study to suggest that DRL agents may exhibit temporal interference effects in a dual-task paradigm similar to those observed in human timing research.

4 Analysis of neural dynamics

An obvious place to start exploring the neural dynamics corresponding to temporal processing was the long short-term memory (LSTM) layer of the DRL agents. This is because, LSTMs, a type of recurrent neural network, are known to capture temporal dependencies in their input [14]. Additionally, in another variant of the DRL agents (not described in this study) where the LSTM layer was removed, the agent, while still able to successfully perform the single task (T), did not exhibit any time-keeping behavior. Instead, the agent simply placed the onion in the oven and repeatedly performed the 'Interact' with oven action until it could take the soup out. Hence, it solely relied on state changes to perform the task. Moreover, studies involving continuous-time recurrent neural networks (CTRNNs) [18] and deep recurrent neural networks (RNNs) [17] have demonstrated that recurrent layers can encode temporal information through biologically plausible mechanisms such as oscillations, ramping activity, and time cells.
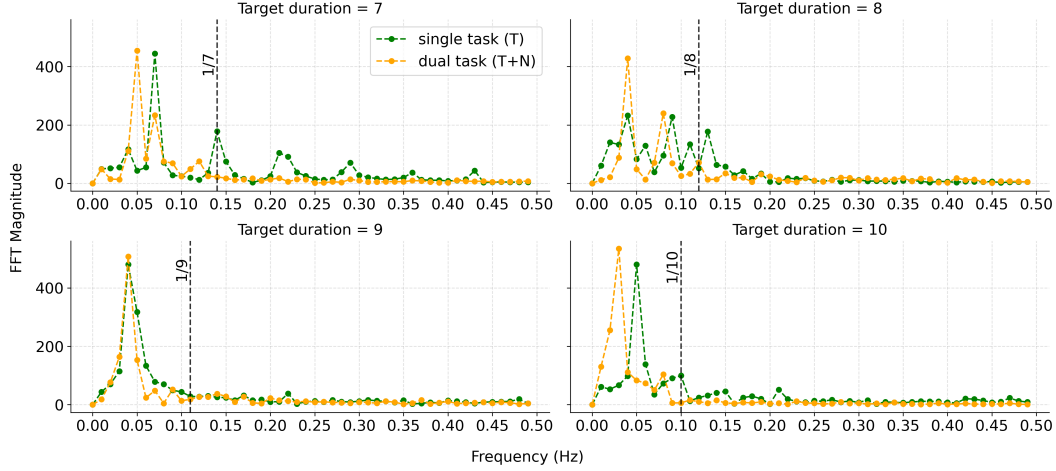
Figure 4: A fast fourier transform (FFT) of the first principal components of the LSTM hidden state activations (across first 100 time steps) for the single task (T) (in green) and the dual task (T+N) (in orange) agents across different durations. The black dotted line marks the target interval frequency.

A principal component analysis (PCA) of the LSTM layer's hidden state activations across first 100 time steps revealed complex oscillatory dynamics with multiple peaks within each cycle (see Appendix). These oscillations were reset at the onset of each new trial (after soup delivery). Thus, it clearly carried information about the start and end of a trial. However, no evidence of a timer or any type of timing mechanism was observed in this analysis. Interestingly, previous work involving a DRL agent performing a dedicated time reproduction task showed counter-like neural activities in the PCA of the LSTM layer [10]. The absence of such a dedicated timing mechanism may be attributed to the timing task being embedded in the soup delivery task and no immediate reward provided for correctly producing the target interval.

A fast fourier transform (FFT) was applied on the first principal components to disentangle the different oscillation patterns (see Figure 4). For target durations 7 and 10, the single task (T) agent exhibited some oscillations with frequencies equal to or greater than the target duration, suggesting accurate timing or underproduction. In contrast, the dual task (T+N) counterparts showed oscillations with frequencies lower than the target duration, suggesting overproduction. However, no clear link can be established between these peaks and an intrinsic time-keeping mechanism. Thus, further analysis is needed to explain the neural dynamics underlying the observed behavioral differences.

5   Discussion

This study aims to contribute to recent research interest of drawing parallels between DNNs and biological systems to facilitate a better understanding of both [1, 16, 9].While such research is extensively carried out in vision and audio, studies on temporal processing remain limited [10, 17]. Prior research has examined time-keeping mechanisms in DRL agents, revealing biologically plausible features such as ramping cells, time cells, or timing biases. The current study adds to this line of work by demonstrating new biologically plausible behavior in DRL agents.

This study has several limitations, a few of them are as follows. First, the reward structure could be improved by using a delayed reward in the dual task setting. This would better align with corresponding human timing studies and help to further verify the biological similarities in the observed behavior. Second, the study was limited to only four target durations, including a wider range could further substantiate the findings. Third, the analysis of neural dynamics focused exclusively on the LSTM layer and was preliminary in nature. Future work could extend this by incorporating other layers from the DRL agent's architecture.

References

[1] David GT Barrett, Ari S Morcos, and Jakob H Macke. Analyzing biological and artificial neural networks: challenges with opportunities for synergy? *Current opinion in neurobiology*, 55:55–64, 2019.

[2] Hamit Basgol, Inci Ayhan, and Emre Ugur. Time perception: A review on psychological, computational, and robotic models. *IEEE Transactions on Cognitive and Developmental Systems*, 14(2):301–315, 2021.

[3] Richard A Block and Ronald P Gruber. Time perception, attention, and memory: a selective review. *Acta psychologica*, 149:129–133, 2014.

[4] Scott W Brown. Attentional resources in timing: Interference effects in concurrent temporal and nontemporal working memory tasks. *Perception & psychophysics*, 59(7):1118–1140, 1997.

[5] Scott W Brown. Timing and executive function: Bidirectional interference between concurrent temporal production and randomization tasks. *Memory & cognition*, 34(7):1464–1471, 2006.

[6] Scott W Brown, Shawn A Collier, and Jill C Night. Timing and executive resources: dual-task interference patterns between temporal production and shifting, updating, and inhibition tasks. *Journal of Experimental Psychology: Human Perception and Performance*, 39(4):947, 2013.

[7] Catalin V Buhusi and Warren H Meck. What makes us tick? functional and neural mechanisms of interval timing. *Nature reviews neuroscience*, 6(10):755–765, 2005.

[8] Micah Carroll, Rohin Shah, Mark K Ho, Tom Griffiths, Sanjit Seshia, Pieter Abbeel, and Anca Dragan. On the utility of learning about humans for human-ai coordination. *Advances in neural information processing systems*, 32, 2019.

[9] Radoslaw M Cichy and Daniel Kaiser. Deep neural networks as scientific models. *Trends in cognitive sciences*, 23(4):305–317, 2019.

[10] Ben Deverett, Ryan Faulkner, Meire Fortunato, Gregory Wayne, and Joel Z Leibo. Interval timing in deep reinforcement learning agents. *Advances in Neural Information Processing Systems*, 32, 2019.

[11] David M Eagleman. Human time perception and its illusions. *Current opinion in neurobiology*, 18(2):131–136, 2008.

[12] C Fortin and R Rousseau. Time estimation as an index of processing demand in memory search. *Perception & Psychophysics*, 42(4):377–382, 1987.

[13] Simon Grondin. Timing and time perception: A review of recent behavioral and neuroscience findings and theoretical directions. *Attention, Perception, & Psychophysics*, 72(3):561–582, 2010.

[14] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[15] Richard B Ivry and John E Schlerf. Dedicated and intrinsic models of time perception. *Trends in cognitive sciences*, 12(7):273–280, 2008.

[16] Nancy Kanwisher, Meenakshi Khosla, and Katharina Dobs. Using artificial neural networks to ask 'why' questions of minds and brains. *Trends in Neurosciences*, 46(3):240–254, 2023.

[17] Dongyan Lin, Ann Zixiang Huang, and Blake Aaron Richards. Temporal encoding in deep reinforcement learning agents. *Scientific Reports*, 13(1):22335, 2023.

[18] Michail Maniadakis, Panos Trahanias, and Jun Tani. Explorations on artificial time perception. *Neural Networks*, 22(5-6):509–517, 2009.

[19] William J Matthews and Warren H Meck. Time perception: the bad news and the good. *Wiley Interdisciplinary Reviews: Cognitive Science*, 5(4):429–446, 2014.

[20] Hugo Merchant, Deborah L Harrington, and Warren H Meck. Neural basis of the perception and estimation of time. *Annual review of neuroscience*, 36(1):313–336, 2013.

[21] Kia Nobre and Jennifer Theresa Coull. *Attention and time*. Oxford University Press, 2010.

[22] Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021.

[23] Warrick Roseboom, Zafeirios Fountas, Kyriacos Nikiforou, David Bhowmik, Murray Shanahan, and Anil K Seth. Activity in perceptual classification networks as a basis for human subjective time perception. *Nature communications*, 10(1):267, 2019.

6 Appendix

6.1 RL environment

The agent state consisted of a spatial array with different integers indicating the different objects. The oven state whether on or off, agent state whether carrying onion or carrying soup were represented as one-hot encoding through different channels of the input array. Thus, while the agent received a change in state each time the oven was on, it did not receive any indication that the onion soup was ready (i.e., the target interval had passed). There was no upper limit on when the agent could take out the soup. Once taken out, the oven state changed to off and the agent input state updated. The agent could take six actions in total. Five of which corresponded to navigating in the grid ('wait', 'up', 'down', 'left' and 'right') and one action to 'interact' with the different objects by standing adjacent to it.

6.2 DRL agent architecture

The agent architecture consisted of a convolutional neural network (CNN) layer designed to process spatial information from the 5x3 grid environment, primarily utilizing 1x1 convolutions. This was followed by a long short-term memory (LSTM) layer comprising 256 hidden units, which captured temporal dependencies. Subsequently, a multilayer perceptron (MLP) with 64 hidden units further processed the extracted features
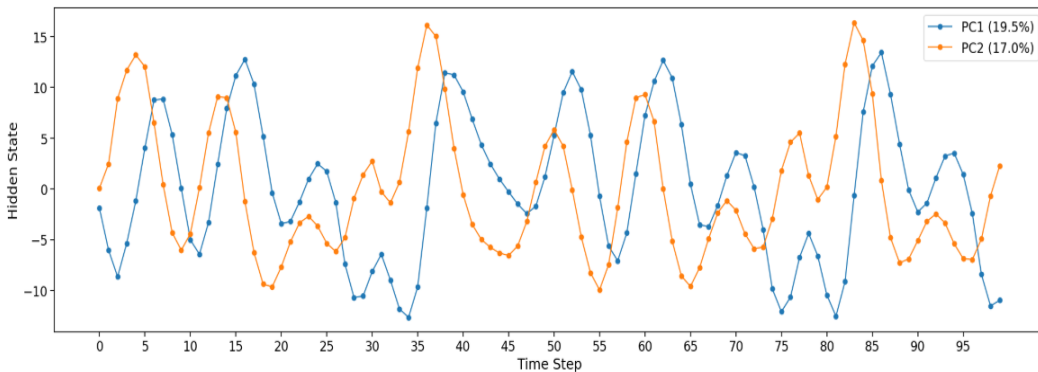
6.3 PCA of LSTM hidden states

Figure 5: PCA of LSTM hidden state activations across 100 time steps for single task (T) agent
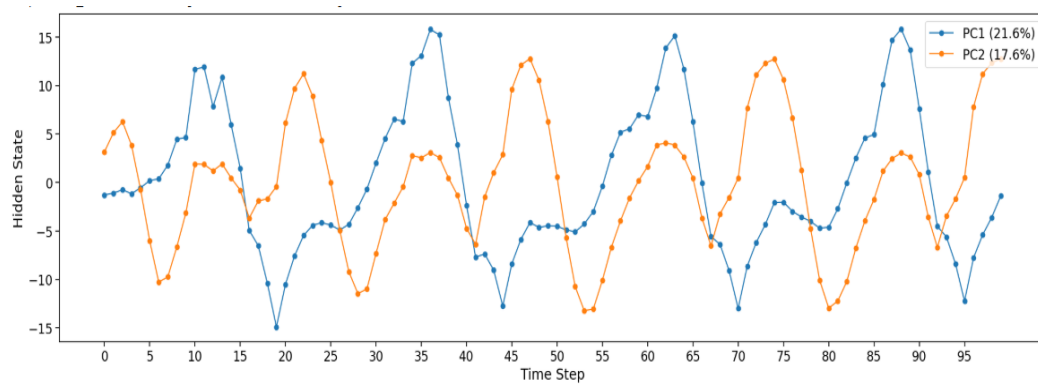
7

Figure 6: PCA of LSTM hidden state activations across 100 time steps for dual task (T+N) agent