Rethinking Teacher-Student Curriculum Learning through the Cooperative Mechanics of Experience

Manfred Diaz
Mila, University of Montreal

Liam Paull
Mila, University of Montreal

Andrea Tacchetti
Google DeepMind

Abstract

Teacher-Student Curriculum Learning (TSCL) is a curriculum learning framework that draws inspiration from human cultural transmission and learning. It involves a teacher algorithm shaping the learning process of a learner algorithm by exposing it to controlled experiences. Despite its success, understanding the conditions under which TSCL is effective remains challenging. In this paper, we propose a data-centric perspective to analyze the underlying mechanics of the teacher-student interactions in TSCL. We leverage cooperative game theory to describe how the composition of the set of experiences presented by the teacher to the learner, as well as their order, influences the performance of the curriculum that are found by TSCL approaches. To do so, we demonstrate that for every TSCL problem, there exists an equivalent cooperative game, and several key components of the TSCL framework can be reinterpreted using game-theoretic principles. Through experiments covering supervised learning, reinforcement learning, and classical games, we estimate the cooperative values of experiences and use value-proportional curriculum mechanisms to construct curricula, even in cases where TSCL struggles. The framework and experimental setup we present in this work represent a foundation that can be used for a deeper exploration of TSCL, shedding light on its underlying mechanisms and providing insights into its broader applicability in machine learning.

1 Introduction

Controlling the sequence of tasks that a learning algorithm is exposed to through curriculum has been shown to potentially enhance learning efficiency [19, 41, 9]. One widely used curriculum framework, known as Teacher-Student Curriculum Learning (TSCL) [32, 49], specifically gives a *teacher algorithm* the ability to control this sequence. While it is commonly understood that presenting tasks in increasing difficulty can improve learning, the underlying dynamics and structure of teacher-student interaction in this context are still relatively unexplored. Very few works have attempted to understand *when*, and *how* TSCL works [45, 79] while most have focused on providing algorithmic improvements to the problem [61, 73, 46, 21]. In this paper, we propose a novel *data-centric* perspective [55] to understand and analyze TSCL algorithms.

We begin by formalizing a general notion of *units of experience* to describe the control objects of the teacher algorithm (that are consumed by the learner). Subsequently, our approach draws inspiration from work on feature attribution [60], data valuation [26, 81] and explainability [48], and leverages tools from cooperative game theory [75, 68] to analyze *how* the compositions of these units impact teacher-student interactions. We show that, for every TSCL problem, there exists an equivalent cooperative game where *units of experience* are players and teacher-student interactions

2nd Workshop on Agent Learning in Open-Endedness (ALOE) at NeurIPS 2023.

approximate a sequential coalition formation process (Sec. 4). As a result, the learning progression objective [65, 57, 32] and the teacher bandit policy [28, 49], two essential components of TSCL, have alternative interpretations as an approximation of player (unit) marginal contribution [77] and a fair allocation mechanism, respectively (Sec. 4.2 & 4.3). Furthermore, because the *order matters* in the case of curriculum learning [41, 9], traditional cooperative game-theoretic arguments produce unintuitive results [56]. Thus, we leverage *generalized cooperative games* and their solution concepts [56, 64] to overcome these limitations and formally extend these data-centric game-theoretic formulations to the curriculum learning setting.

To demonstrate the predictive power and range of problems where this game-theoretic and data-centric interpretation of TSCL applies, we build an experimental setting that evaluates the prospect of cooperation among *units of experience* in problems spanning supervised learning (SL), reinforcement learning (RL), and classical games (Sec. 5). These experiments simulate ordered and unordered coalition formation processes and approximate the cooperative games we developed to describe TSCL. For every problem, we estimate units *a priori* value (e.g., Shapley or Nowak & Radzik values) and demonstrate that these *a priori* values, although expensive to compute [16], are useful proxies to find curricula. To this end, we design unordered and ordered value-proportional curriculum mechanisms inspired by value-proportional allocations [7]. In most settings, the unordered mechanism fails to find a reasonable curriculum, demonstrating the unsuitability of traditional game-theoretic tools for the TSCL problem. However, the ordered mechanism consistently finds an optimal or near-optimal ordering (i.e., a curriculum) even when TSCL fails (Sec. 5.5). To understand what impacts the ability of TSCL in those settings, we leverage another cooperative game-theoretic tool, namely, *measures of interactions* [31, 62], and in particular the Value of a Player to other Player (vPoP) [33], to quantify positive, neutral, or negative pairwise interactions among units. We show that in settings with considerable unit interference (i.e., negative interactions) TSCL is unable to produce useful curricula.

2 Preliminaries

2.1 Cooperative Game Theory

Cooperative Games. Cooperative games model problems where players interact to maximize collective gain [63]. In a (traditional) cooperative game in characteristic function form among a set of players $\mathbf{U}$, denoted by $\mathcal{G} = \langle \mathbf{U}, v \rangle$, the characteristic function $v : 2^{\mathbf{U}} \to \mathbb{R}$ associates to each coalition $\mathbf{C} \in 2^{\mathbf{U}}$, belonging to the powerset $2^{\mathbf{U}}$, a real number that represents the benefits produced by the players in $\mathbf{C}$ acting jointly. In a cooperative game, a solution concept represents a mechanism that produces allocation vectors $\phi \in \mathbb{R}^{|\mathbf{U}|}$ [70]. Particularly, *Shapley's value* [68] allocates to each player $\mathbf{u} \in \mathbf{U}$ its average marginal contribution $v(\mathbf{C} + \mathbf{u}) - v(\mathbf{C})$ to coalitions $\mathbf{C} \subseteq \mathbf{U}$, where $\mathbf{u} \in \mathbf{U} - \mathbf{C}$

$$\phi(\mathbf{u}) = \sum_{\mathbf{C}:\mathbf{u} \notin \mathbf{C}} \frac{|\mathbf{C}|!(|\mathbf{U}| - |\mathbf{C}| - 1)!}{|\mathbf{U}|!} \left[v(\mathbf{C} + \mathbf{u}) - v(\mathbf{C}) \right] \tag{1}$$

and uniquely satisfies the axioms of *efficiency*, *null-player*, *symmetry*, and *linearity* which are generally considered to be properties of a fair allocation mechanism [74].

Generalized Cooperative Games. When the order on which players join determines coalitional worth, traditional cooperative games and their solution concepts (e.g., Shapley's value) may produce unintuitive allocations [56]. In these games, the generalized characteristic function $v : \mathcal{P}(2^{\mathbf{U}}) \to \mathbb{R}$ assigns to every ordered coalition $\mathbf{C} \in \mathcal{P}(2^{\mathbf{U}})$ in the powerset of permutations $\mathcal{P}(2^{\mathbf{U}})$ its worth if members join in the permutation order. Nowak & Radzik [56] and Sanchez & Bergantiños [64] extended Shapley's work and propose solution concepts for these generalized cooperative games. We focus on the former due to its intuitive formulation

$$\phi_{\text{NR}}(\mathbf{u}) = \frac{1}{|\mathbf{U}|!} \sum_{\substack{\mathbf{C} \in \mathcal{P}(2^{\mathbf{U}}) \\ \mathbf{C}:\mathbf{u} \notin \mathbf{C}}} \left[v(\mathbf{C} : \mathbf{u}) - v(\mathbf{C}) \right] \tag{2}$$

that averages, for all ordered coalitions $\mathbf{C} \in \mathcal{P}(2^{\mathbf{U}})$ where the unit $\mathbf{u} \in \mathbf{U}$ is appended last, its marginal contribution to the newly formed ordered coalition $\mathbf{C} : \mathbf{u}$.

Measures of Interactions. A measure of interactions [31, 62] computes players' influences on other players' outcomes. In particular, we leverage the *value of a player to another player* (vPoP) [33].

For the games above, vPoP constructs a matrix whose entries $\phi(\mathbf{u}_i, \mathbf{u}_j) \in \mathbb{R}$ measure the influence player $\mathbf{u}_i$ exerts over player $\mathbf{u}_j$. It measures how the Shapley value of a unit changes in the absence of another. More precisely,

$$\phi(\mathbf{u}_i, \mathbf{u}_j) = \sum_{\substack{\mathbf{C} \subseteq \mathbf{U} \\ \mathbf{u}_i, \mathbf{u}_j \in \mathbf{C}}} \frac{(|\mathbf{U}| - |\mathbf{C}|)(|\mathbf{C}| - 1)!}{|\mathbf{U}|!} \left[\phi(\mathbf{u}_j, \mathbf{C}) - \phi(\mathbf{u}_j, \mathbf{C} - \mathbf{u}_i)\right] \tag{3}$$

where $\phi(\mathbf{u}_j, \mathbf{C})$ is the Shapley value of unit $\mathbf{u}_j$ (Eq. 1) in the cooperative game restricted to players in $\mathbf{C}$. This matrix marginal $\phi(\mathbf{u}_i) = \sum_j \phi(\mathbf{u}_i, \mathbf{u}_j)$ corresponds to each player's Shapley value. We extend vPoP to games in generalized characteristic function form by applying Eq. 3 *mutatis mutandis* using Nowak & Radzik [56] value to provide an ordered pairwise interaction metric $\phi_{\text{NR}}(\mathbf{u}_i, \mathbf{u}_j)$.

2.2 Bandit Algorithms

Multi-armed bandit algorithms provide a solution to problems decision-making under uncertainty [28, 43] where, at each interaction, a decision must be made about with arm $\mathbf{u} \in \mathbf{U}$ must be pulled. We are particularly interested on action-value based algorithms that maintain empirical value estimates $q_k(\mathbf{u})$ computed as

$$q_k(\mathbf{u}) \approx \frac{1}{N_k^{\mathbf{u}}} \sum_{i=1}^{k-1} r(\mathbf{u}_i) \mathbb{I}_{\mathbf{u}_i = \mathbf{u}} \tag{4}$$

and that estimate the average reward received by the algorithm in the iterations $N_k^{\mathbf{u}} \leq k$ where the $\mathbf{u}$-arm has been pulled. Bandit algorithms, like the ones Graves et al. [32] and Matiisen et al. [49] use in their work, transform the estimated average contributions into arms interactions by deriving from estimated values a Boltzmann policy $\tau_k \in \Delta(\mathbf{U})$ such that the probability of interaction is proportional to the value estimates:

$$\tau_k(\mathbf{u}) \propto \mathcal{B}(q_k(\mathbf{u})) = \frac{e^{\frac{q_k(\mathbf{u})}{T}}}{\sum_{\mathbf{u}'} e^{\frac{q_k(\mathbf{u}')}{T}}} \tag{5}$$

More sophisticated approaches (e.g., the EXP3 [2] used in our experiments) account for other factors, like recency, bias, stochasticity, or non-stationarity [43].

3 Experience to Control

The TSCL framework commonly operates under the assumption that tasks presented to a learning algorithm can influence its learning dynamics. Modern iterative learning algorithms process tasks in discrete units. For instance, SL and RL algorithms operate over instances and transitions, respectively. But also, collections of these elementary units, such as batches or episodes, datasets or *environments*, or more generally benchmarks or environment suites describe a hierarchy of aggregations of experience. Henceforth, we utilize the term ***unit of experience*** for referring to any collection of discrete units that a teacher algorithm can use to control the dynamics of the learner algorithm.

Example 3.1. For an analysis, we may define a *unit of experience* as the set of instances of class in a SL classification problem. For example, in the *MNIST* dataset [44], there may be *ten units of experience*, namely, classes ZERO, ONE, TWO, . . . , NINE.

The *units of experience* abstraction indistinctly applies to supervised or reinforcement learning problems. On either paradigm, any iterative learning algorithm is a controllable system whose control inputs are units of experience.

Example 3.2. There are four control inputs in mini-batch gradient descent [30]: the mini-batch $\{x_1, \ldots, x_B\}$, the loss function ℓ, the parameters θ, and the learning rate η such that:

$$\theta_{k+1} = \theta_k - \eta \nabla_{\theta_k} \sum_{i=1}^{B} \ell(\theta_k, x_i)$$

A TSCL-style algorithm, as presented in Alg. 1, solves a data-centric control problem and thus, we adopt a data-centric perspective to perform a systematic investigation of its components.

1: **procedure** GENTSCL
2: **inputs** policy: π_0, algorithm: $\mathcal{L}$, units: $\mathbf{U}$, metric: $\mathcal{J}$
3: *teacher*: $\tau_0 \in \Delta(\mathbf{U})$, *targets*: $\bar{\mathbf{U}}$, budget: K
4: **for** $k = 1 \dots K$ **do**
5: $\mathbf{u}_k \sim \tau_k(\mathbf{u})$
6: $\pi_k \sim \mathcal{L}(\pi_{k-1}, \mathbf{u}_k)$
7: $r_k \leftarrow \mathcal{J}(\pi_k, \bar{\mathbf{U}}) - \mathcal{J}(\pi_{k-1}, \bar{\mathbf{U}})$ ▷ learning progression
8: $\tau_{k+1} \leftarrow$ UPDATERULE$(\tau_k, \mathbf{u}_k, r_k)$ ▷ a bandit algorithm
9: **end for**
10: **output:** π_K
11: **end procedure**

Algorithm 1: In Experience-based Teacher-Student Curriculum Learning (TSCL), the *learner* algorithm $\mathcal{L}(\pi_{k-1}, \mathbf{u}_k)$ is a *black-box* system (line 6) controlled by a *teacher* algorithm through units drawn with probability $\mathbf{u}_k \sim \tau_k(\mathbf{u})$. The *learner* output, at each iteration k, is policy or model π_k whose performance is measured by a *metric function* $\mathcal{J}$ that quantifies model's performance on a set of evaluation units $\bar{\mathbf{U}}$. The teacher's objective is to maximize the cumulative learning progression reward (line 7). For the *teacher*'s UPDATERULE, we focus on multi-armed bandit learning (see Sec. 2.2).

4 The Cooperative Mechanics of Experience

The ideal teacher-student interaction mechanics assume that the *learner* monotonically increases its performance on the target task. We conjecture that a prerequisite for this idealistic curriculum learning dynamics [49] to occur within TSCL-style algorithms is that experience (or data) presented to the *learner* should not interfere with each other. In other words, *units of experience* should interact cooperatively. We explain how this cooperative mechanics may emerge among units by examining the history of teacher-student interactions, the reward function, and the bandit selection policy from a cooperative game-theoretic perspective.

4.1 The Mechanics of Coalition Formation

We establish a cooperative game where each unit of experience $\mathbf{u} \in \mathbf{U}$ is a *player*. Next, we interpret the history of $k \leq K$ teacher-student interactions $\mathbf{H}_k = \{\mathbf{u}_1, \dots, \mathbf{u}_k\}$ through their empirical frequencies $p_k(\mathbf{u}) \in \Delta_{\mathbf{U}}$ which form unit vectors that lie in the $|\mathbf{U}|$-probability simplex $\Delta(\mathbf{U})$. The effective support (i.e., non-zero probabilities) determine an unordered coalition (i.e., a set) $\mathbf{C}_k \subseteq \mathbf{U}$ (see Faigle [20], Chapter 8), formed by the units presented to the *learner* up to interaction $k \leq K$. We study this interpretation through a cooperative game in characteristic function form (Sec. 2.1).

Example 4.1. **(Example 3.1 cont'd)** In the *class-as-unit* equivalence on MNIST, an unordered training coalition, e.g., the two-unit coalition $\mathbf{C} = \{\text{ZERO}, \text{NINE}\}$, describes teacher-student interactions limited to instances from those *classes*.

Next, we note that the outcome of a coalition's work is the policy or model π_k. Thus, estimating the performance of the policy π_k through the metric function $\mathcal{J}$ is akin to approximating the *characteristic function* $v(\mathbf{C}_k)$ (Alg. 1, line 7). Moreover, these approximations are conditioned on an evaluation (or target) unit $\bar{\mathbf{u}} \in \bar{\mathbf{U}}$. We model the *target-task* and *multiple-task* settings [32] where *units of experience* should increase *learner* performance on an evaluation unit (e.g., a task, or an environment) or on multiple evaluation units (e.g., a set of tasks or environments). Consequently, every notion of coalitional worth is conditional on the evaluation units, thus generating a space of cooperative games.

Definition 4.1. (**TSCL Cooperative Games**) Let $\mathbf{U}$ denote a set of *units of experience* $\mathbf{u} \in \mathbf{U}$ and $\bar{\mathbf{U}}$ a set *evaluation units* $\bar{\mathbf{u}} \in \bar{\mathbf{U}}$. Every evaluation coalition $\bar{\mathbf{C}} \in 2^{\bar{\mathbf{U}}}$ induces a parameterized characteristic function $v_{\bar{\mathbf{C}}}(\mathbf{C}_k) \in \mathbb{R}$ whose value measures the worth of a coalition $\mathbf{C}_k$ when the members of $\bar{\mathbf{C}}$ are the *evaluation units*. Therefore, the TSCL-family of algorithms operate over a parameterized space of cooperative games:

$$\mathcal{G}\langle \mathbf{U}, \cdot \rangle = \left\{ \langle \mathbf{U}, v_{\bar{\mathbf{C}}} \rangle \mid \bar{\mathbf{C}} \subseteq \bar{\mathbf{U}} \right\}$$

comprising $2^{|\mathbf{U}|} \times 2^{|\bar{\mathbf{U}}|}$ possible games and where the *target-task* (i.e., $\bar{\mathbf{C}} = \bar{\mathbf{u}}$) and the *multiple-tasks* (i.e., $\bar{\mathbf{C}} = \bar{\mathbf{U}}$) settings are special cases.

Example 4.2. If a *learner* algorithm is presented with units from $\mathbf{C} = \{\text{ZERO}, \text{NINE}\}$ on MNIST, the following condition is expected to hold:

$$v_{\bar{\mathbf{C}}=\{\text{ZERO},\text{NINE}\}}(\mathbf{C}) > v_{\bar{\mathbf{C}}=\{\text{ZERO},\text{ONE}\}}(\mathbf{C})$$

4.2 Marginal Contributions to Learning

The notions of coalitions and coalitional worth above induce a game-theoretic interpretation of the learning progression reward. At any iteration $k \leq K$, this reward $r(\mathbf{u}_k) \in \mathbb{R}$ (Alg. 1, line 7) measures the improvement in policy performance after the *teacher* presents a unit $\mathbf{u}_k$ to the *learner* algorithm that produces a new policy $\pi_k \sim \mathcal{L}(\pi_{k-1}, \mathbf{u}_k)$. Thus, we can restate this reward in terms of a game in characteristic function form:

$$r(\mathbf{u}_k) = v(\mathbf{C}_k) - v(\mathbf{C}_{k-1}) = v(\mathbf{C}_{k-1} + \mathbf{u}_k) - v(\mathbf{C}_{k-1}) \tag{6}$$

and note its equivalence to computing the marginal contribution (see Sec. 2 and Eq. 1) of aggregating the unit of experience $\mathbf{u}_k = \mathbf{u}$ to the existing coalition $\mathbf{C}_{k-1}$.

4.3 A Fair Allocation Mechanism

A principle of fair attribution in cooperative games is that players get assigned values proportional to their expected marginal contribution. We note that under the learning progression objective, a bandit action-value estimate $q_k(\mathbf{u})$ (Eq. 4) approximates every unit's (or arm's) average marginal contribution after k interactions. Moreover, as discussed in Sec. 2.2, multi-arm bandit algorithms may transform action-values through a Boltzmann projection (Eq. 5) that converts the value estimates into units' probabilities of interaction with the *learner* (i.e., the (stochastic) policy $\tau_k(\mathbf{u})$). Consequently, the units that up to interaction $k \leq K$ have produced larger increases on *learner* performance would be allocated larger fractions of the remaining $K - k$ interactions.

Thus, a multi-armed bandit teacher implements a fair allocation mechanism that computes units' values by approximating their average marginal contributions and converts these approximations into the currency-like utility of the TSCL games, namely, interactions with the learner.

5 An Experiment on The Prospect of Cooperation

We introduce an experimental setting that, for any given set of *units of experience* that may be given to the *teacher* algorithm, approximates the *a priori* units' values in a series of experiments that include problems in supervised learning, reinforcement learning, and classical games. Through the *prospect prior*, as we dub these experiments, we study how units' interactions impact TSCL's prospects to find useful curricula.

5.1 A Simulation of Cooperation

We simulate two coalition formation processes where *units of experience* (e.g., classes, environments, or opponents) in each coalition fairly share a finite interaction budget $K \in \mathbb{N}$. First, to simulate a traditional cooperative game and approximate its characteristic function (Sec. 2), we draw at each interaction $k \leq K$ a unit $\mathbf{u}_k \in \mathbf{C}$ with uniform probability $\tau_{\mathbf{C}}(\mathbf{u}_k) \propto \frac{1}{|\mathbf{C}|}$, from a coalition $|\mathbf{C}|$, and present it to a *learner* with algorithm $\mathcal{L}$. We repeat this procedure for every unordered coalition of units $\mathbf{C} \in 2^{\mathbf{U}}$. The uniform distribution reflects our ignorance on units' values before simulating their effect on the *learner*. Then, to study order in curriculum, we simulate a generalized cooperative game and approximate its corresponding characteristic function, a *unit* $\mathbf{u} \in \mathbf{C}$ is continually presented to the *learner*, for $\lfloor K/|\mathbf{C}| \rfloor$ interactions, in its permutation order on an ordered coalition $\mathbf{C}$. We repeat this procedure for every $\mathbf{C} \in \mathcal{P}(2^{\mathbf{U}})$. Similarly to the unordered case, we select an ordered equipartition of interactions to reflect ignorance about *a priori* units' values.

Coalitional Worth. In both simulations, and for every coalition, we obtain a model π_K at the end of the K interactions. The resulting policy or model is not biased with respect to any *evaluation unit or coalition*. Thus, we leverage the established equivalence between policy or model performance and (conditional) coalitional worth to estimate the value of every parameterized characteristic function and form the space of traditional or generalized cooperative games. By estimating every coalition worth, we have the complete specification of a cooperative game.

Example 5.1. (Example 4.1 cont'd) Assume a budget of $K = 100$ interactions and a subset (coalition) of classes from MNIST, for instance, units (classes) ZERO and NINE. In the simulation of a traditional game, for a coalition $\mathbf{C} = \{\text{ZERO}, \text{NINE}\}$, we uniformly draw instances from each unit with probability $\tau(\mathbf{u}) = \frac{1}{2}$. For a coalition $\mathbf{C} = [\text{ZERO}, \text{NINE}]$ in a generalized game, instances from unit ZERO are presented for the first $k = 50$ iterations followed by $k = 50$ instances from NINE.

5.2 Value Approximations

The central quantity for the cooperative solution concepts we introduced in Sec. 2 is a unit's marginal contribution. We design the *prospect prior* experiment such that adding a unit $\mathbf{u}$ to a coalition $\mathbf{C}$ has the effect of reducing the learner algorithm's interactions with the existing units in the coalition by keeping the budget K fixed, regardless of the coalition size. In the traditional cooperative game simulation, the probability of drawing any unit $\mathbf{u}_k$ in $\mathbf{C}$ gets reduced from $p_{\mathbf{C}}(\mathbf{u}_k) = \frac{1}{|\mathbf{C}|}$ to $p_{\mathbf{C}+\mathbf{u}}(\mathbf{u}_k) = \frac{1}{|\mathbf{C}|+1}$. Similarly, in the generalized game simulation adding a unit $\mathbf{u}_k$ to a coalition $\mathbf{C}$ reduces the number of interactions of units in $\mathbf{C}$ from $\lfloor \frac{K}{|\mathbf{C}|} \rfloor$ to $\lfloor \frac{K}{|\mathbf{C}+1|} \rfloor$. Consequently, in either *prospect prior* simulation, a unit $\mathbf{u}$ marginal contribution $v(\mathbf{C} + \mathbf{u}) - v(\mathbf{C})$ measures the change in performance produced by increasing interactions with $\mathbf{u}$ while reducing interactions of the existing units in $\mathbf{C}$.

Example 5.2. (Example 5.1 cont'd) In a traditional game simulation on MNIST, a marginal contribution such as $v(\{\text{ZERO}, \text{NINE}\}) - v(\{\text{ZERO}\})$ measures the change in *learner* performance produced by exchanging approximately $K/2$ interactions with unit ZERO for interactions with NINE. However, in the generalized game simulation, the same expression measures the change in performance produced by exchanging K interactions with unit $\{\text{ZERO}\}$ for $K/2$ with ZERO first (pre-training) followed by $K/2$ with NINE (fine-tuning).

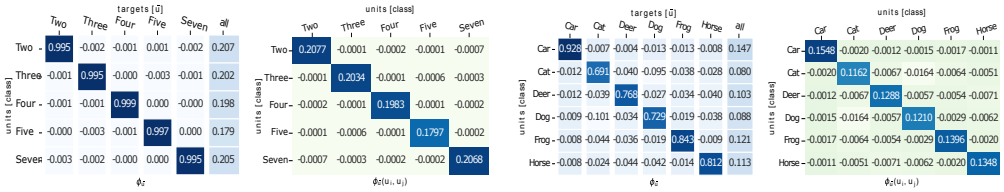
Thus, the solution concepts for each simulated cooperative game, namely, the *Shapley value* for traditional games (Eq. 1) and the *Nowak & Razik's value* for generalized games (Eq. 2) estimate a unit's average marginal contribution to learning, thus capturing its helpfulness or cooperativeness.

5.3 Supervised Classification

The first setting we examine is inspired by our running examples on MNIST. There, we considered instances aggregated in classes as *units of experience*. The main benefit of this toy example is that it is straightforward to produce ground truth information about units' interactions by training a model on the complete dataset (e.g., for 200 epochs), extracting the model's confusion matrix on validation, and identifying the *top-k* most confused pairs of classes. We applied the same general procedure to CIFAR10 [39]. On MNIST, we selected the five classes TWO, THREE, FOUR, FIVE and SEVEN belonging to top-three most confused pairs (see Appendix A, Fig. 5a), grouped their instances into five *units of experience*, and conduct the *prospect prior* simulations for a traditional cooperative game. We did similarly for CIFAR10 six classes with larger pairwise confusion errors on validation, namely, CAR, CAT, DEER, DOG, FROG and HORSE (see Appendix A, Fig. 5b).

Units' Values. For either MNIST or CIFAR10, each unit's Shapley value estimated from the traditional cooperative game simulations correctly matches the ground truth information. In the target-unit setting, where each *unit of experience* is also used as evaluation unit, each *unit of experience* (or class) matching the evaluation unit (or class) has the largest *Shapley value*, as depicted in Fig. 2a (first *five* targets) for MNIST and Fig. 2c (first *six* targets) for CIFAR10. For instance, on MNIST, unit $\mathbf{u} = \text{TWO}$ has the largest *Shapley value* $\phi(\text{TWO}) = 0.995$ when the evaluation unit is $\bar{\mathbf{u}} = \text{TWO}$. We observed a similar effect on CIFAR10. Then, for the *all-units* setting, every unit's *Shapley value* is approximately equal on both MNIST and CIFAR10 (Fig. 2a and 2c, *all* column), matching the intuition that, conditional on an *all* units evaluation, every *unit of experience* should be equally valuable. These results confirm the *prospect prior*'s ability to correctly estimate units' values.

Measures of Interactions. We also computed the vPoP measure (see Sec. 2.1, Eq. 3) to verify whether its decomposition of *Shapley values* into pairwise interaction values correctly identifies the most confused pairs of classes. For both MNIST and CIFAR10, their respective vPoP matrices, displayed in Fig. 2b and 2d, provide a good approximation to the ground-truth pairwise interactions extracted from the confusion matrices. For instance, on MNIST the *units* TWO and SEVEN have the lowest interaction value $\phi(\text{TWO}, \text{SEVEN}) = \phi(\text{SEVEN}, \text{TWO}) = -0.007$ which corresponds to largest



(a) (b) (c) (d)

Figure 2: We validated the **prospect prior** using the *class-as-a-unit* analogy on MNIST and CIFAR10. In Figures (a) and (c), each column represents units' Shapley values $\phi(\mathbf{u})$ in each cooperative game parameterized by a target-unit $\bar{\mathbf{u}}$ and the target coalition of *all* units. In Figures (b) and (d), we present the vPoP decomposition matrix (Eq. 3) measuring the pairwise interaction values $\phi(\mathbf{u}_i, \mathbf{u}_j)$ among units in the *all-units* target.

entry $M(2, 7) = 20$ of the confusion matrix (see Appendix A, Fig. 5a). Also, in CIFAR10 the units DOG and CAT have the lowest interaction value $\phi(\text{DOG}, \text{CAT}) = -0.0164$ coinciding with the most confused classes $M(\text{DOG}, \text{CAT}) = 66$ on validation (see Appendix A, Fig. 5b). However, we note that, for instance, on MNIST confusion matrix $M(2, 7) \neq M(7, 2)$ and, similarly, on CIFAR10's $M(\text{DOG}, \text{CAT}) \neq M(\text{CAT}, \text{DOG})$. Nevertheless, we interpret these values as good proxies for units *negative, positive, or neutral* pairwise interactions.

Our formulation is not limited to the toy supervised classification setting. We also demonstrate its general applicability to problems in RL and classical games[1].

5.4 Other Learning Paradigms

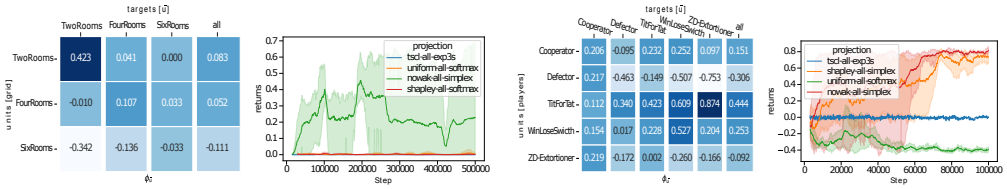
Reinforcement Learning. We investigate the MINIGRID-ROOMS [15] set of three environments, namely, TWOROOMS, FOURROOMS, and SIXROOMS for which it is *folk knowledge* that an optimal curriculum exists. We apply the *prospect prior* simulation of a generalized game where we consider each environment a *unit of experience*. As a *learner* algorithm, we used PPO [66] with an interaction budget of $K = 500,000$ steps, and estimated, from the outcome of these simulations, the *Nowak & Radzik* values (Sec. 2, Eq. 2), conditioned on the every environment, and on a uniform distribution over *all*, as evaluation targets. In Fig. 3a we show that the *Nowak & Radzik* values we estimate match *folk knowledge*. First, there is no requirement for environments other than TWOROOMS as the only positive value $\phi_{\bar{\mathbf{u}}}(\text{TWOROOMS}) = 0.423$ correctly measures. Then, for FOURROOMS, the values of $\phi_{\bar{\mathbf{u}}}(\text{TWOROOMS}) = 0.041$ and $\phi_{\bar{\mathbf{u}}}(\text{FOURROOMS}) = 0.107$ indicate that both environments are required. And finally, environments values of $\phi_{\bar{\mathbf{u}}}(\text{FOURROOMS}) = 0.03$ and $\phi_{\bar{\mathbf{u}}}(\text{SIXROOMS}) = 0.03$ indicate that both are needed for solving SIXROOMS.

Classical Games. We introduce an experimental setting, the *Adversarial Sparse Iterated Prisoner's Dilemma* (A-SIPD), that utilizes the Prisioner's Dilemma [23, 4] classical two-player game as a base but in a more challenging *sparse* and iterated version where at the end of a finite number of interactions (e.g., 200 steps), a *win-draw-loss* reward is given to the learner if it accrues more cumulative *payoff* than its opponent. Opponents are drawn from a *population* of five well-known strategies: ALWAYSCOOPERATE, ALWAYSDEFECT, WINSTAYLOSESWITCH, TITFORTAT [5] and a ZERODETERMINANT strategy [34]. We apply the *prospect prior* simulation of a *generalized game* where we consider each *opponent* a *unit of experience*. As a *learner* algorithm, we used PPO [66] with a budget of $K = 100,000$ interactions. We estimated the *Nowak & Radzik* values (Sec. 2, Eq. 2), conditioned on each opponent, and on a uniform mixture over *all*, as evaluation targets. The results we present in Fig. 3c show that playing uniquely against TITFORTAT is sufficient across all evaluation targets, including the strongest opponents, ALWAYSDEFECT and ZERODETERMINANT. This result contrasts with *folk knowledge* in population-based training (e.g, playing against the population's *Nash strategy* [54]). We defer to Appendix B a more in-depth discussion of this finding.

5.5 Prospect Values and Curriculum

Inspired by value-proportional allocations [7], we developed a mechanism that turns units' values into interactions with the *learner* by projecting any value vector $\phi(\mathbf{u}) \in \mathbb{R}^{|\mathbf{U}|}$ onto vectors $\tau(\mathbf{u}) \in \Delta_{\mathbf{U}}$

[1]Details to reproduce these experiments are provided in Appendix A.



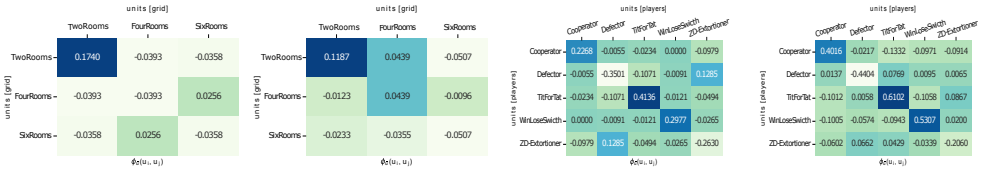
(a) *Nowak & Radzik Values* (b) Learning Curves [*all*] (c) *Nowak & Radzik Values* (d) Learning Curves [*all*]

MINIGRID-ROOMS A-SIPD

Figure 3: *Nowak & Radzik* values *(a, c)* conditional on each *single-unit* and the *multiple-units* (all) evaluations, and the *learning curves (b,d)* for our mechanisms and TSCL.

in a $|\mathbf{U}|$-*simplex*. We investigate two of these projections[2], the *Boltzmann* or *softmax* projection, commonly used in *multi-arm bandit* algorithms (see Sec. 2.2, Eq. 5), and an *Euclidean* projection [11] that projects to zero any unit with negative value. When values $\phi(\mathbf{u})$ are *Shapley values*, the projected vectors are used as pre-computed *teacher* policies (i.e., probability distributions), mimicking TSCL's interactions but fixed *a-priori* knowledge of units' value. However, when values $\phi(\mathbf{u})$ are *Nowak & Radzik*, vectors $\tau \in \Delta_{\mathbf{U}}$ are used as *ordered compositional vectors* [1] that represent ordered fractions of K interactions. Thus, we construct a pre-computed *teacher* policy that, first orders units by their *Nowak & Radzik values*, projects the ordered values onto $\tau \in \Delta_{\mathbf{U}}^{K}$, and presents the *i-th ranked unit* $\mathbf{u}_i \in \mathbf{U}$ to the *learner* for the number of interactions indicated by $\tau_i \in \mathbb{N}$. This mechanism preserves the ordered values captured by *Nowak & Radzik*'s solution concept.

We compare these *value-proportional mechanisms* with the multi-arm bandit approach to TSCL using EXP3 [2, 32]. Fig. 3 shows, for the *all-units* evaluation, that for both MINIGRID-ROOMS and A-SIPD the *teacher* policies obtained from the Euclidean projection of *Nowak & Radzik* values (i.e, *nowak-all-simplex*) consistently produce learner-induced policies outperforming those produced by TSCL (i.e, *tscl-all-exp3s*), and the other *mechanisms*. These results highlight TSCL's inability to produce effective curriculum in the presence of units that have non-cooperative interactions. Fig. 4 presents the vPoP decomposition of *Shapley* and *Nowak & Radzik* values conditioned on the *all-units* evaluation. In both settings, the interactions measured from the *Shapley*-based decomposition produce lower (negative) value than those obtained with *Nowak & Radzik*. As we show through Sec. 4 and Sec. 5, we take the *Shapley*-based interaction values as fair approximations of TSCL interactions, and thus they provide a data-centric explanation to TSCL failures (see Appendix C). On the other hand, the *Nowak & Radzik*-based values explain the *nowak-all-simplex* mechanism success, along with the elimination, through the *Euclidean* projection, of negatively-valued units[3].



(a) Shapley-vPoP (b) Nowak&Radzik-vPoP (c) Shapley-vPoP (d) Nowak&Radzik-vPoP

Figure 4: The vPoP decomposition of *Shapley's* and *Nowak & Radzik values*, conditioned on the *all-units* evaluation target, for the MINIGRID-ROOMS *(a, b)* and *A-SIPD (c, d)* problem settings.

6 Related Work

Since the seminal work of Elman [19], curriculum learning has become a fundamental problem in machine learning [41, 9]. The TSCL framework we study here was concurrently introduced by Graves et al. [32] and Matiisen et al. [49]. Most follow-up works have generally focused on algorithmic innovations [53, 76, 71]. However, Wu et al. [79] empirically explore the questions on *when* and *how* curriculum learning works but lack any formal grounding. Meanwhile, Lee et al. [45]

[2]Bachrach et al. [7] use a *linear* projection unable to handle negative marginal contributions.

[3]This elimination strategies are employed effectively by data valuation techniques [81]

explore TSCL interactions through *catastrophic forgetting* in neural networks [50]. However, the *data-centric* perspective we introduced is less explored.

Moreover, our work continues the cross-pollination tradition between machine learning and game theory [13]. Recently, game theory has fuelled several areas of machine learning [29, 25, 14]. In particular, cooperative game theory has influenced analysis on feature attribution [60], data valuation [26, 27, 81, 82], and explainability [48]. The *feature-as-a-player* analogy used in literature on explainability inspired the notion of *units of experience* we introduced in Sec. 3.

Recasting TSCL through cooperative game theory highlights its connections to active learning [67], multitask [12] and continual learning [58]. As Graves et al. [32], Matiisen et al. [49] note, TSCL is connected to *active learning* [67] through the idea of *sampling experience* according to a prioritization mechanism. Our work also relates TSCL to multitask [12] and continual learning [58, 52] through solution concepts for generalized cooperative games [56, 64] which may warrant further investigation on those areas. Likewise, *measures of interaction* like vPoP may provide a formalism for task relatedness and its effects on learning [72, 83, 22].

7 Limitations

The simulations we computed in our experiments would be hard to carry beyond a handful of units. It is well-established that cooperative solution concepts are computationally hard [16, 18]. Although better approximations are possible [81, 51], we do not explore them in this work. Consequently, we do not consider the *prospect prior* experiments and the *value-proportional curriculum mechanism* as algorithmic replacements of TSCL. The *data-centric* approach we present studies the limits imposed on TSCL-style algorithms by the (non)cooperative mechanics among units of experience. However, we acknowledge that the mechanics of units' interactions also affect other aspects of TSCL. These aspects may include, for instance, the *teacher*'s credit-assignment problem [28] or neural networks learning and forgetting dynamics (e.g., Lee et al. [45]). We control for these factors by keeping them constant in our experiments (see Appendix A) but do not undertake their analysis here. Our work is a starting point for more thorough explorations of TSCL and curriculum learning, their underlying mechanisms and broader applicability in machine learning.

8 Conclusions & Future Work

We reexamined TSCL through the lens of cooperative game theory. By drawing inspiration from work on data valuation, feature attribution and explainability, we provide a novel data-centric perspective that re-frames several of its components through alternative cooperative game-theoretic interpretations. Our experiments confirmed the appropriateness of studying TSCL-style under this framework and highlights the impact of units cooperative mechanics on this problem. However, we only began to unveil the potential of allocation mechanisms, solution concepts, and measures of interactions to explain some fundamental aspects of TSCL and hope our work inspires an influx of novel game-theoretic approaches to the problem. Future work would explore more theoretically-grounded analysis of this problem through the connection between convex games [69] and super(sub)modularity in discrete combinatorial optimization [17, 6, 38] and the extension to continuous set of units through values of non-atomic games [3].

References

[1] J Aitchison. The statistical analysis of compositional data. *Journal of the Royal Statistical Society*, 44(2):139–160, January 1982. 8

[2] Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multi-armed bandit problem. *SIAM J. Comput.*, 32(1):48–77, jan 2003. ISSN 0097-5397. doi: 10.1137/S0097539701398375. URL https://doi.org/10.1137/S0097539701398375. 3, 8, 16, 17

[3] R. J. Aumann and L. S. Shapley. *Values of Non-Atomic Games*. Princeton University Press, 1974. URL http://www.jstor.org/stable/j.ctt13x149m. 9

[4] R Axelrod and W D Hamilton. The evolution of cooperation. *Science*, 211(4489):1390–1396, March 1981. 7

[5] Robert Axelrod. The emergence of cooperation among egoists. *The American political science review*, 75(2):306–318, 1981. 7, 17

[6] Francis Bach. Learning with submodular functions: A convex optimization perspective. November 2011. 9

[7] Yoram Bachrach, Richard Everett, Edward Hughes, Angeliki Lazaridou, Joel Z Leibo, Marc Lanctot, Michael Johanson, Wojciech M Czarnecki, and Thore Graepel. Negotiating team formation using deep reinforcement learning. *Artificial intelligence*, 288(103356):103356, November 2020. 2, 7, 8

[8] David Balduzzi, Marta Garnelo, Yoram Bachrach, Wojciech Czarnecki, Julien Perolat, Max Jaderberg, and Thore Graepel. Open-ended learning in symmetric zero-sum games. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 434–443. PMLR, 2019. 17

[9] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ICML '09, pp. 41–48, New York, NY, USA, 2009. Association for Computing Machinery. ISBN 9781605585161. URL https://doi.org/10.1145/1553374.1553380. 1, 2, 8

[10] Lilian Besson. SMPyBandits: an Open-Source Research Framework for Single and Multi-Players Multi-Arms Bandits (MAB) Algorithms in Python. Online at: github.com/SMPyBandits/SMPyBandits, 2018. URL https://github.com/SMPyBandits/SMPyBandits/. Code at https://github.com/SMPyBandits/SMPyBandits/, documentation at https://smpybandits.github.io/. 16, 17

[11] Mathieu Blondel, Akinori Fujino, and Naonori Ueda. Large-Scale multiclass support vector machine training via euclidean projection onto the simplex. In *2014 22nd International Conference on Pattern Recognition*, pp. 1289–1294, August 2014. 8

[12] Rich Caruana. Multitask learning. *Machine learning*, 28:41–75, 1997. 9

[13] Nicolo Cesa-Bianchi and Gabor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006. doi: 10.1017/CBO9780511546921. 9

[14] Michael Chang, Sid Kaushik, S Matthew Weinberg, Tom Griffiths, and Sergey Levine. Decentralized reinforcement learning: Global Decision-Making via local economic transactions. In Hal Daumé Iii and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1437–1447. PMLR, 2020. 9

[15] Maxime Chevalier-Boisvert, Lucas Willems, and Suman Pal. Minimalistic gridworld environment for gymnasium, 2018. URL https://github.com/Farama-Foundation/Minigrid. 7, 16

[16] Xiaotie Deng and Christos H. Papadimitriou. On the complexity of cooperative solution concepts. 19:257–266, 1994. ISSN 0364-765X. 2, 9

[17] Shaddin Dughmi. Submodular functions: Extensions, distributions, and algorithms. a survey. December 2009. 9

[18] Edith Elkind, Leslie Ann Goldberg, Paul W Goldberg, and Michael Wooldridge. On the computational complexity of weighted voting games. *Annals of mathematics and artificial intelligence*, 56(2):109–131, June 2009. 9

[19] J L Elman. Learning and development in neural networks: the importance of starting small. *Cognition*, 48(1):71–99, July 1993. 1, 8

[20] Ulrich Faigle. *Mathematical Game Theory*. 2022. 4

[21] Dieqiao Feng, Carla P Gomes, and Bart Selman. A novel automated curriculum strategy to solve hard sokoban planning instances. October 2021. 1

[22] Christopher Fifty, E. Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. Efficiently identifying task groupings for multi-task learning. *Advances in neural information processing systems*, 2021. ISSN 1049-5258. URL https://www.semanticscholar.org/paper/b7df7b89c9ce9050fc7102458e98f0e8f673374d. 9

[23] Merrill M. Flood. *Some Experimental Games*. RAND Corporation, Santa Monica, CA, 1952. 7

[24] Marta Garnelo, Wojciech Marian Czarnecki, Siqi Liu, Dhruva Tirumala, Junhyuk Oh, Gauthier Gidel, Hado van Hasselt, and David Balduzzi. Pick your battles: Interaction graphs as population-level objectives for strategic diversity. In *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '21, pp. 1501–1503, Richland, SC, 2021. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450383073. 17

[25] Ian Gemp, Brian McWilliams, Claire Vernade, and Thore Graepel. Eigengame: {PCA} as a nash equilibrium. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=NzTU59SYbNq. 9

[26] Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. April 2019. 1, 9

[27] Amirata Ghorbani, James Zou, and Andre Esteva. Data shapley valuation for efficient batch active learning. April 2021. 9

[28] J C Gittins. Bandit processes and dynamic allocation indices. *Journal of the Royal Statistical Society*, 41(2):148–164, January 1979. 2, 3, 9

[29] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K.Q. Weinberger (eds.), *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014. URL https://proceedings.neurips.cc/paper_files/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf. 9

[30] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016. 3

[31] Michel Grabisch and Marc Roubens. An axiomatic approach to the concept of interaction among players in cooperative games. *International Journal of Game Theory*, 28(4):547–565, Nov 1999. ISSN 0020-7276. 2

[32] Alex Graves, Marc G Bellemare, Jacob Menick, Rémi Munos, and Koray Kavukcuoglu. Automated curriculum learning for neural networks. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1311–1320. PMLR, 2017. 1, 2, 3, 4, 8, 9, 16, 17

[33] Kjell Hausken and Matthias Mohr. The value of a player in n-person games. *Social choice and welfare*, 18(3):465–483, 2001. ISSN 0176-1714. 2

[34] Christian Hilbe, Martin A. Nowak, and Karl Sigmund. Evolution of extortion in iterated prisoner's dilemma games. *Proceedings of the National Academy of Sciences of the United States of America*, 110(17):6913–6918, Apr 2013. ISSN 0027-8424. 7, 17

[35] Shengyi Huang, Rousslan Fernand Julien Dossa, Chang Ye, Jeff Braga, Dipam Chakraborty, Kinal Mehta, and João G.M. Araújo. Cleanrl: High-quality single-file implementations of deep reinforcement learning algorithms. *Journal of Machine Learning Research*, 23(274):1–18, 2022. URL `http://jmlr.org/papers/v23/21-1342.html`. 16, 17

[36] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL `http://arxiv.org/abs/1412.6980`. 15, 16, 17

[37] Vince Knight, Owen Campbell, Marc, T J Gaffney, Eric Shaw, Vsn Reddy Janga, Nikoleta Glynatsi, James Campbell, Karol M Langner, Sourav Singh, Julie Rymer, Thomas Campbell, Jason Young, MHakem, Geraint Palmer, Kristian Glass, Daniel Mancia, edouardArgenson, Martin Jones, kjurgielajtis, Yohsuke Murase, Sudarshan Parvatikar, Melanie Beck, Cameron Davidson-Pilon, Marios Zoulias, Adam Pohl, Paul Slavin, Timothy Standen, Aaron Kratz, and Areeb Ahmed. Axelrod-Python/Axelrod: v4.12.0, October 2021. 17

[38] Andreas Krause and Carlos Guestrin. Submodularity and its applications in optimized information gathering. *ACM Trans. Intell. Syst. Technol.*, 2(4):1–20, July 2011. 9

[39] Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, April 2009. 6

[40] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 15

[41] Kai A Krueger and Peter Dayan. Flexible shaping: how learning in small steps helps. *Cognition*, 110(3):380–394, March 2009. 1, 2, 8

[42] Marc Lanctot, Vinicius Zambaldi, Audrunas Gruslys, Angeliki Lazaridou, Karl Tuyls, Julien Pérolat, David Silver, and Thore Graepel. A unified game-theoretic approach to multiagent reinforcement learning. *Advances in neural information processing systems*, 30, 2017. 17

[43] Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, July 2020. 3

[44] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010. URL `http://yann.lecun.com/exdb/mnist/`. 3, 15

[45] Sebastian Lee, Sebastian Goldt, and Andrew M. Saxe. Continual learning in the teacher-student setup: Impact of task similarity. *International Conference on Machine Learning*, 2021. URL `https://www.semanticscholar.org/paper/57db7f24f15150ef7ea0db1fed20e6ee752792ec`. 1, 8, 9

[46] Rui Liu, Tianyi Wu, and Barzan Mozafari. Adam with bandit sampling for deep learning. October 2020. 1

[47] Siqi Liu, Luke Marris, Daniel Hennes, Josh Merel, Nicolas Heess, and Thore Graepel. NeuPL: Neural population learning. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=MIX3fJkl_1`. 17

[48] Scott M. Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pp. 4768–4777, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964. 1, 9

[49] Tambet Matiisen, Avital Oliver, Taco Cohen, and John Schulman. Teacher-Student curriculum learning. *IEEE transactions on neural networks and learning systems*, 31(9):3732–3740, September 2020. 1, 2, 3, 4, 8, 9

[50] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In Gordon H Bower (ed.), *Psychology of Learning and Motivation*, volume 24, pp. 109–165. Academic Press, January 1989. 9

[51] Rory Mitchell, Joshua Cooper, Eibe Frank, and Geoffrey Holmes. Sampling permutations for shapley value estimation. *Journal of machine learning research: JMLR*, 23(43):1–46, 2022. 9

[52] Martin Mundt, Yongwon Hong, Iuliia Pliushch, and Visvanathan Ramesh. A wholistic view of continual learning with deep neural networks: Forgotten lessons and the bridge to active and open world learning. *Neural networks: the official journal of the International Neural Network Society*, 160:306–336, March 2023. 9

[53] Sanmit Narvekar, Bei Peng, Matteo Leonetti, Jivko Sinapov, Matthew E Taylor, and Peter Stone. Curriculum learning for reinforcement learning domains: a framework and survey. *Journal of machine learning research: JMLR*, 21(1):7382–7431, June 2022. 8

[54] J F Nash. Equilibrium points in N-Person games. *Proceedings of the National Academy of Sciences of the United States of America*, 36(1):48–49, January 1950. 7, 17

[55] Andrew Ng. Mlops: From model-centric to data-centric ai. `https://www.youtube.com/watch?v=06-AZXmwHjo`, 2021. 1

[56] Andrzej S Nowak and Tadeusz Radzik. The shapley value for n-person games in generalized characteristic function form. *Games and economic behavior*, 6(1):150–161, January 1994. 2, 3, 9

[57] Pierre-Yves Oudeyer, Frédéric Kaplan, and Verena V Hafner. Intrinsic motivation systems for autonomous mental development. *IEEE transactions on evolutionary computation*, 11(2): 265–286, 2007. 2

[58] German I Parisi, Ronald Kemker, Jose L Part, Christopher Kanan, and Stefan Wermter. Continual lifelong learning with neural networks: A review. *Neural networks: the official journal of the International Neural Network Society*, 113:54–71, May 2019. 9

[59] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pp. 8024–8035. Curran Associates, Inc., 2019. URL `http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf`. 15, 16, 17

[60] Roma Patel, Marta Garnelo, Ian Gemp, Chris Dyer, and Yoram Bachrach. Game-theoretic vocabulary selection via the shapley value and banzhaf index. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2789–2798, Online, June 2021. Association for Computational Linguistics. 1, 9

[61] Rémy Portelas, Cédric Colas, Katja Hofmann, and Pierre-Yves Oudeyer. Teacher algorithms for curriculum learning of deep RL in continuously parameterized environments. October 2019. 1

[62] Ariel Procaccia, Nisarg Shah, and Max Tucker. On the structure of synergies in cooperative games. *Proceedings of the AAAI Conference on Artificial Intelligence*, 28(1), Jun 2014. ISSN 2374-3468. doi: 10.1609/aaai.v28i1.8812. URL `https://ojs.aaai.org/index.php/AAAI/article/view/8812`. 2

[63] Alvin E Roth (ed.). *The Shapley value: essays in honor of Lloyd S. Shapley*. Cambridge University Press, October 1988. 2

[64] Estela Sanchez and Gustavo Bergantiños. On values for generalized characteristic functions. *Operations-Research-Spektrum*, 19(3):229–234, September 1997. 2, 9

[65] J. Schmidhuber. *A possibility for implementing curiosity and boredom in model-building neural controllers*. The MIT Press, 1991. ISBN 9780262256674. 2

[66] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 7, 16, 17

[67] Burr Settles. Active learning literature survey. *Machine learning*, 15(2):201–221, 2010. 9

[68] Lloyd S Shapley. A value for N-Person games. Technical report, RAND Corporation, 1952. 1, 2

[69] Lloyd S Shapley. Cores of convex games. *International Journal of Game Theory*, 1(1):11–26, December 1971. 9

[70] Martin Shubik. Game theory models and methods in political economy. In Kenneth J Arrow and Michael D Intriligator (eds.), *Handbook of Mathematical Economics*, volume 1, pp. 285–330. Elsevier, 1981. 2

[71] Petru Soviany, Radu Tudor Ionescu, Paolo Rota, and Nicu Sebe. Curriculum learning: A survey. *International journal of computer vision*, 130(6):1526–1565, June 2022. 8

[72] Trevor Standley, Amir R Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? May 2019. 9

[73] Matteo Turchetta, Andrey Kolobov, Shital Shah, Andreas Krause, and Alekh Agarwal. Safe reinforcement learning via curriculum induction. *Advances in neural information processing systems*, 33:12151–12162, 2020. 1

[74] René van den Brink and Gerard van der Laan. Axiomatizations of the normalized banzhaf value and the shapley value. *Social Choice and Welfare*, 15(4):567–582, 1998. ISSN 01761714, 1432217X. URL `http://www.jstor.org/stable/41106281`. 2

[75] John Von Neumann and Oskar Morgenstern. *Theory of games and economic behavior*. Princeton University Press, Princeton, NJ, 1944. 1, 17

[76] Xin Wang, Yudong Chen, and Wenwu Zhu. A survey on curriculum learning. *IEEE transactions on pattern analysis and machine intelligence*, 44(9):4555–4576, September 2022. 8

[77] Robert J Weber. Probabilistic values for games. In A E Roth (ed.), *Essays on the Shapley Value and Its Applications*, pp. 101–119. Cambridge University Press, 1988. 2

[78] Michael P. Wellman. Methods for empirical game-theoretic analysis. In *Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2*, AAAI'06, pp. 1552–1555. AAAI Press, 2006. ISBN 9781577352815. 17

[79] Xiaoxia Wu, Ethan Dyer, and Behnam Neyshabur. When do curricula work? *International Conference on Learning Representations*, 2020. 1, 8

[80] Omry Yadan. Hydra - a framework for elegantly configuring complex applications. Github, 2019. URL `https://github.com/facebookresearch/hydra`. 15

[81] Tom Yan and Ariel D Procaccia. If you like shapley then you'll love the core. *Proceedings of the ... AAAI Conference on Artificial Intelligence. AAAI Conference on Artificial Intelligence*, 35(6):5751–5759, May 2021. 1, 8, 9

[82] Shunyu Yao, Mo Yu, Yang Zhang, Karthik Narasimhan, Joshua Tenenbaum, and Chuang Gan. Linking emergent and natural languages via corpus transfer. In *International Conference on Learning Representations (ICLR)*, 2022. 9

[83] Yipeng Zhang, Tyler L Hayes, and Christopher Kanan. Disentangling transfer and interference in Multi-Domain learning. *ArXiv*, 2021. 9

A Prospect Prior Experiments Details

We provide for all problems, models or policies architectures, algorithm hyperparameters, and other reproducibility details[4]. All models and architectures are implemented with PYTORCH [59], are configured using HYDRA [80], and fit on a workstation equipped with a 16 GB *NVIDIA RTX A4000* GPU, 32 GB of RAM, and 32 CPU cores.

A.1 Supervised Learning

MNIST. We trained a model on the MNIST [44] supervised *10-digits* classification task. Specification of the model architecture and hyper-parameters selection are provided in Table 1.

Hyperparameter	Value
optimizer	ADAM [36]
learning-rate	10^{-4}
betas	$(0.9, 0.999)$
eps	10^{-8}
batch-size	4
epochs	200
shuffle	*Yes*

Model
CONV2D$(32, 3, 1)$
RELU()
CONV2D (64, 3, 1)
RELU()
MAXPOOL2D (2, 2)
DROPOUT(0.25)
FLATTEN()
LINEAR(9216, 128)
RELU()
DROPOUT(0.5)
LINEAR(128, 10)

Table 1: Details on the learning algorithm hyperparameters (*left*) and model architecture (*right*) used in the MNIST [44] experiments. Model components and the optimizer are provided by PYTORCH [59]. These details remained constant throughout the rest of the experiments with MNIST.

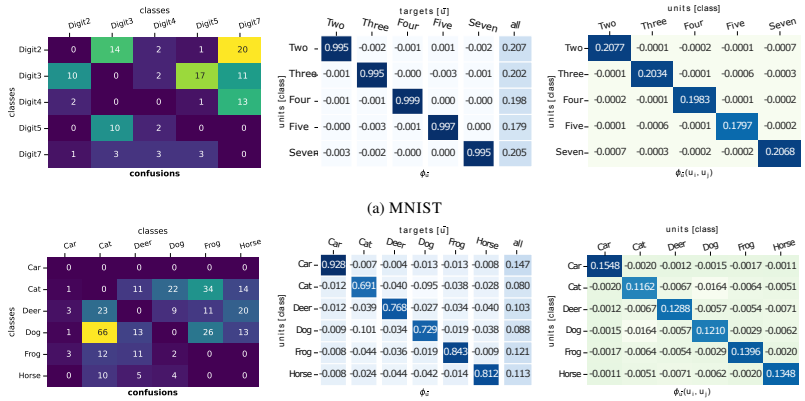
CIFAR10. The experiments on CIFAR10 [40] follow the same setting as those on MNIST. We similarly trained a model on the supervised *10-classes* task. Specification of the model architecture and hyperparameter selection are provided in Table 2.

Hyperparameter	Value
optimizer	ADAM [36]
learning-rate	10^{-4}
betas	$(0.9, 0.999)$
eps	10^{-8}
batch-size	4
epochs	200
shuffle	*Yes*

Model
CONV2D (3, 6, 5)
RELU()
MAXPOOL2D (2, 2)
CONV2D (6, 16, 5)
RELU()
MAXPOOL2D (2, 2)
FLATTEN()
LINEAR(400, 120)
RELU()
LINEAR(120, 84)
RELU()
LINEAR(84, 10)

Table 2: Details on the learning algorithm hyperparameters (*left*) and model architecture (*right*) used in the CIFAR10 [40] experiments. Model components and the optimizer are provided by PYTORCH [59]. These details remained constant throughout the rest of the experiments with CIFAR10.

[4]Regardless, we plan to release the complete source code of all our experiments.



(a) MNIST

(b) CIFAR10

Figure 5: The *class-as-a-unit* analogy applied to MNIST (a) and CIFAR10 (b) served as our ground truth. For each problem, we derived the Shapley's value from the precomputed priors (*left*) [Eq. 1] on each *cooperative game* (Sec. 5). Our results verify that units values on the *target-unit* settings *approximately* ordered the most confused pairs of classes. For instance, digits 2 & 7 in MNIST, or *dog* & cat in CIFAR10. When the target is *all* classes, the vPoP decomposition (*right*) also (Sec. 2.1) identifies *interfering* pairs.

A.2 Reinforcement Learning

MINIGRID ROOMS. We utilized a sequence of TWOROOMS, FOURROOMS, and SIXROOMS *gridworlds* provided on MINIGRID [15] as *units of experience*. As the learning algorithm, we trained for $500,000$ steps a PPO [66] agent, whose implementation we derived from CLEANRL [35]. Policy and actor-critic architecture, with shared backbone, as well as other PPO hyperparameters details are presented in Table 3. For the TSCL experiments, we leveraged *Exp3S* [2] implementation from Besson [10] with default hyperparameters $\alpha = 10^{-5}$ and $\gamma = 0.05$, as defined in Graves et al. [32].

Hyperparameter	Value
optimizer	ADAM [36]
learning-rate	0.0025
annealing	*Yes*
num-steps	128
total-timesteps	$500,000$
seeds	5
gamma	0.99
GAE-lambda	0.95
num-minibatches	4
update-epochs	4
advantage-normalization	*Yes*
clip-value-loss	*Yes*
clip-coeff	0.2
entropy-coeff	0.01
vf-coeff	0.5
max-grad-norm	0.5
target-kl	*No*

Actor	Critic
CONV2D(16, 2, 2)	
RELU()	
MAXPOOL2D(2, 2)	
CONV2D(16, 32, 2, 2)	
RELU()	
CONV2D(16, 64, 2, 2)	
RELU()	
LINEAR$(64, 64)$	LINEAR$(64, 64)$
TANH()	TANH()
LINEAR$(64, 7)$	LINEAR$(64, 1)$

Table 3: Details on the PPO hyperparameters (*left*) and *actor-critic* architecture (*right*) used in the MINIGRID-ROOMS [15] experiments. Policy and critic components, and the optimizer, are provided by PYTORCH [59]. Implementation and default hyperparameters are derived from CLEANRL [35]. These details remained constant throughout the rest of the experiments with MINIGRIDROOMS.

A.3 Populations & Games.

Adversarial SIPD. In our more challenging sparse and iterated version of Prisoner's Dilemma, at the end of 200 interactions, inspired by Axelrod's competition [5]. A *win-draw-loss* reward $r = \{-1, 0, 1\}$ is given to a learning player if it beats a fixed opponent. Opponents are drawn from a population of five well-known strategies: *always cooperate*, *always defect*, *win-stay-lose-switch*, *tit-for-tat*, and a *zero-determinant strategy* [5, 34, 37]. We trained for 500 episodes (or $100,000$ steps) a PPO [66] agent adapted from CLEANRL [35] default implementation. Policy and actor-critic architecture, **without** shared backbone, as well as other PPO hyperparameters details are presented in Table 4. For the TSCL experiments, we leveraged *Exp3S* [2] implementation from Besson [10] with default hyperparameters $\alpha = 10^{-5}$ and $\gamma = 0.05$, as defined in Graves et al. [32].

Hyperparameter	Value
optimizer	ADAM [36]
learning-rate	0.0025
annealing	*Yes*
num-steps	128
timesteps	$100,000$
seeds	5
gamma	0.99
GAE-lambda	0.95
minibatches	4
epochs	4
advantage-norm	*Yes*
clip-value-loss	*Yes*
clip-coeff	0.2
entropy-coeff	0.01
vf-coeff	0.5
max-grad-norm	0.5
target-kl	*No*

Actor	Critic
LINEAR$(2, 64)$	LINEAR$(2, 64)$
ORTHOINIT$()$	ORTHOINIT$()$
TANH$()$	TANH$()$
LINEAR$(64, 64)$	LINEAR$(64, 64)$
ORTHOINIT$()$	ORTHOINIT$()$
TANH$()$	TANH$()$
LINEAR$(64, 2)$	LINEAR$(64, 1)$

Table 4: Details on the PPO hyperparameters (*left*) and *actor-critic* architecture (*right*) used in the ADVERARIAL-SIPD experiments. Policy and critic components, and the optimizer, are provided by PYTORCH [59]. Implementation and default hyperparameters are derived from CLEANRL [35]. These details remained constant throughout the rest of the experiments.

B TSCL and Population-based Training

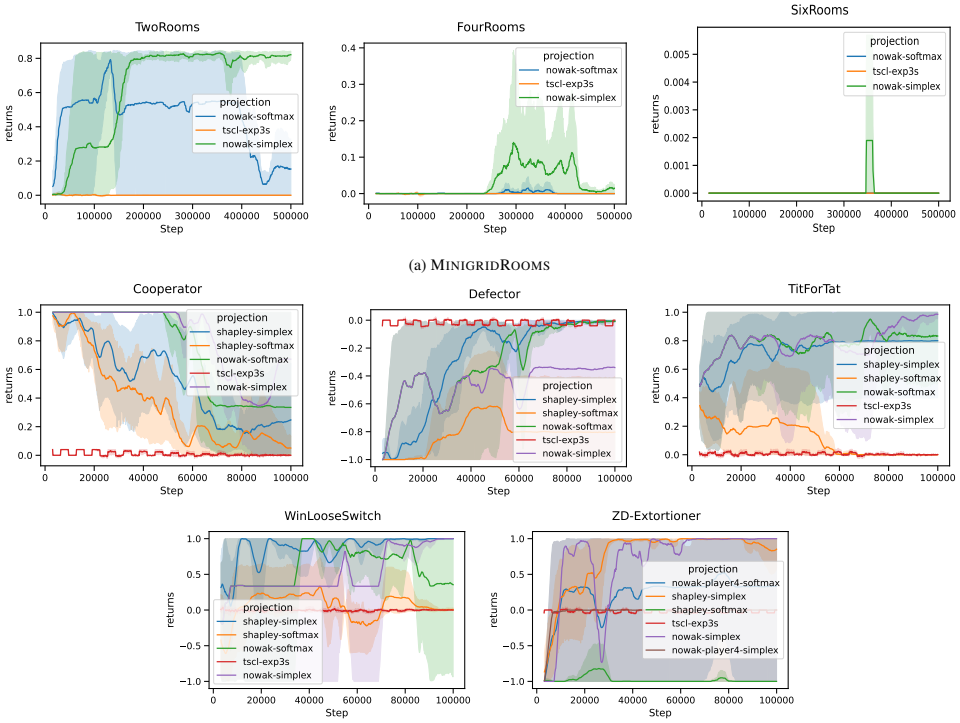
These results in some sense contradict what population-based training approaches prescribe as curriculum learning [42, 8, 24, 47]. Generally, *meta-strategy solvers* for population-based training leverage tools from non-cooperative game theory [75] to find, for instance, the mixed *Nash equilibrium* [54] of the *empirical meta-game* [78] played by the population of opponents. In this sense, we may also understand TSCL as a *cooperative meta-strategy solver* that prioritizes among a fixed population of (non-learning) opponents (units of experience) those that improve the learning progression of a single learning player against one or more opponents of the same population.

In the sparse and iterated version of Prisoner's Dilemma that we introduced, the *Defector* strategy remains to be the (empirical) game *Nash equilibrium*. However, the ordered prospect prior results in Fig. 3c show that when evaluated on the population Nash $\bar{\mathbf{u}} =$ *Defector*, the largest *Nowak & Radzik* value corresponded to the *TitForTat* strategy $\phi_{\bar{\mathbf{u}}}(\textit{TitForTat}) = 0.34$. Playing against the *TitForTat* strategy remains to be the optimal solution across all evaluation targets, meaning that *TitForTat* is the best proxy opponent to beat and reach the Nash equilibrium strategy.

We can interpret this result from two perspectives. First, it could indicate that the sparse, iterated, and overtly adversarial version of the game we constructed is a more complicated problem than the original, and the Nash equilibrium, the *Defector* strategy, is a stronger opponent. However, these result may also indicate that a cooperative approach to meta-strategy selection may improve performance in some scenarios.

C Extended Experiments Results

C.1 Value-Proportional Curriculum

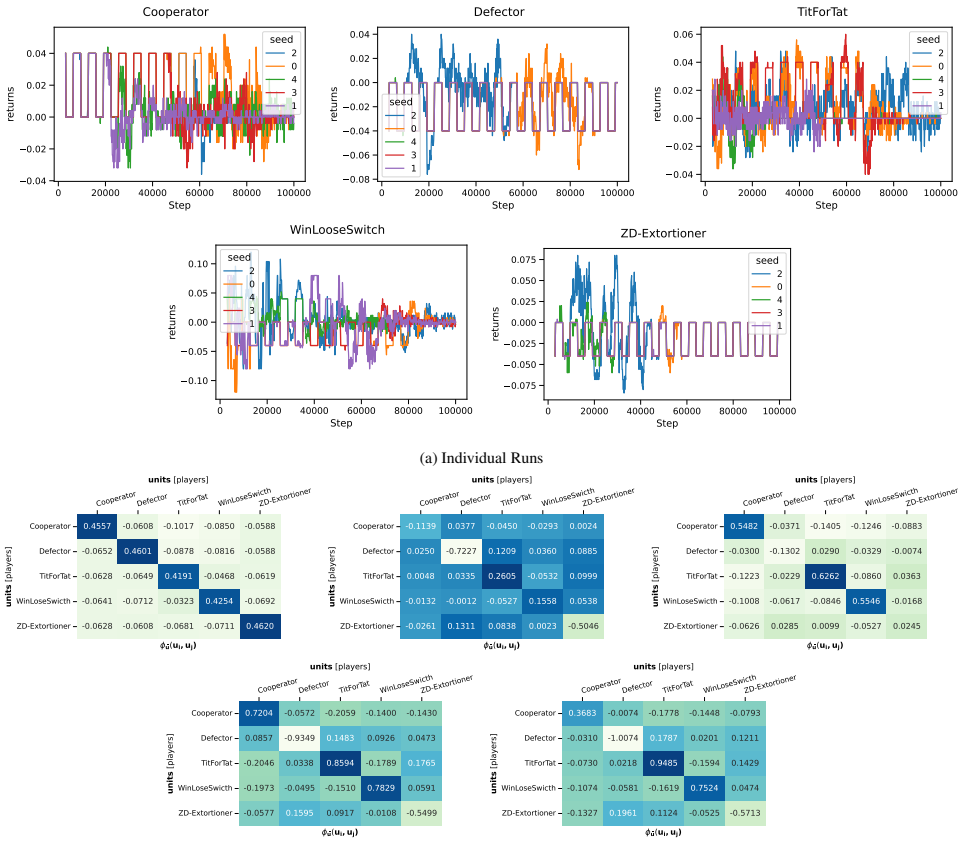


(a) MINIGRIDROOMS

(b) ADVERSARIAL-SIPD

Figure 6: We also investigated the *prior-proportional curriculum* in the *target-unit* setting. For each target unit, we allocate to each training unit interactions proportional to their pre-computed values for each target. For the ADVERSARIAL-SIPD and MINIGRID-ROOMS controlled their learning dynamics by presenting the units according to *unordered* and *ordered* mechanisms in **??**. On each task, the *value-proportional curriculum* derived from the *prospect priot* outperforms TSCL *(tscl-*-exp3s)*. We further investigate the reason for TSCL failures on this scenario.

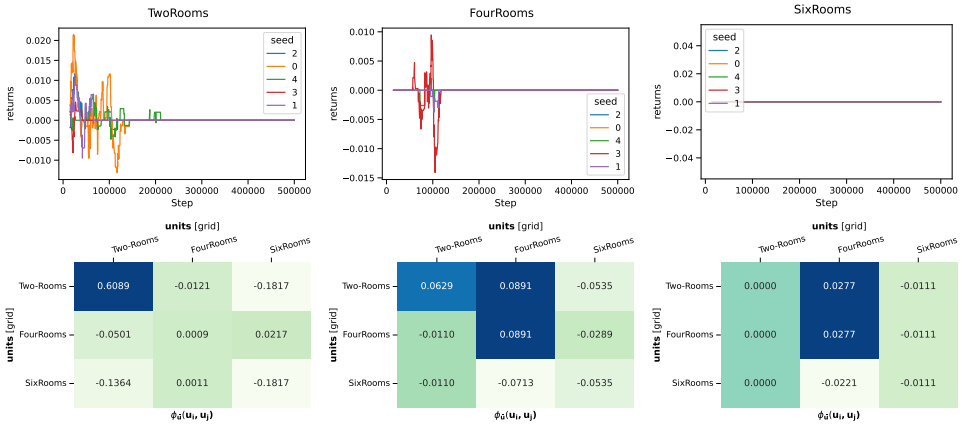
C.2 TSCL Failures



(a) Individual Runs

(b) vPoP Metric

Figure 7: To understand the failure modes of TSCL on Adversarial-SIPD, we represented the individual runs (i.e., each of the five seeds) on every target unit. TSCL *(tscl-*-exp3s)* (top row) is extremely brittle, unstable, and generally not robust to units interference. We surmise that these failures are related to the *exploration-exploitation* dilemma. Exploratory steps presenting a negatively-valued unit are hard to overcome (forgetting dynamics). This issue requires further investigation, and we defer it to future work.



(a) MinigridRooms

Figure 8: We found that TSCL presents a similar problem in MinigridRooms. When actions (units) need to be almost deterministically drawn for several steps, and other actions (units) have negative interference with the target, TSCL is unable to find a stable and robust solution to the p roblem.