

A Significance Testing

This section presents the results of the one-way ANOVA tests on fixed-Bobs evaluation as mentioned in Section 7.1 (Table 6) and the results of the two-sample one-tailed t -Tests on resource allocation comparison as mentioned in Section 7.2 (Table 7).

| Scenario | <i>Chicken</i> | | <i>Pure Coordination</i> | | <i>Prisoners Dilemma</i> | | <i>Stag Hunt</i> | |
|-----------------|---------------------|------------------------|--------------------------|------------------------|--------------------------|------------------------|---------------------|------------------------|
| | <i>F</i> -statistic | <i>p</i> -value | <i>F</i> -statistic | <i>p</i> -value | <i>F</i> -statistic | <i>p</i> -value | <i>F</i> -statistic | <i>p</i> -value |
| <i>Small</i> | 8.338 | 3.46×10^{-4} | 238.9 | 5.19×10^{-51} | 72.48 | 1.01×10^{-23} | 97.51 | 2.83×10^{-29} |
| <i>Medium</i> | 29.03 | 1.25×10^{-11} | 129.0 | 2.77×10^{-35} | 288.7 | 1.90×10^{-56} | 201.0 | 2.83×10^{-46} |
| <i>Large</i> | 49.55 | 8.16×10^{-18} | 67.61 | 1.53×10^{-22} | 75.46 | 1.99×10^{-24} | 132.2 | 7.40×10^{-36} |
| <i>Obstacle</i> | 6.115 | 2.70×10^{-3} | 5.495 | 4.84×10^{-3} | 43.89 | 3.31×10^{-16} | 20.80 | 7.72×10^{-9} |

Table 6: One-way ANOVA tests on fixed-Bobs evaluation. Each ANOVA is summarized with an F -statistic and a p -value. *Result:* Reported ANOVAs confirm that SP, PP3, and PP5 have significant effects on individual rewards in all four specified scenarios under four different environments.

| Scenario | t -statistic | p -value |
|--------------------------|----------------|------------------------|
| <i>Chicken</i> | 3.7649 | 1.050×10^{-4} |
| <i>Pure Coordination</i> | 5.3175 | 1.211×10^{-7} |
| <i>Prisoners Dilemma</i> | 0.7624 | 2.233×10^{-1} |
| <i>Stag Hunt</i> | 2.6498 | 4.297×10^{-3} |

Table 7: Two-Sample One-tailed t -Tests on resource allocation comparison. Each test is summarized with an t -statistic and a p -value. *Result:* Reported tests suggest that except for the Prisoners Dilemma, LoI-guided heuristic allocation has a significant advantage over uniformly allocated cases under the same total resource budget cap.

B Training

This section presents the hyperparameters used for training agents. We train agents on MeltingPot substrates registered in Ray RLlib. We specify one PPO policy per agent. The hyperparameters are reported in Table 8.

| Setting | Value |
|-------------------------------|----------|
| # rollout workers | 2 |
| rollout fragment length | 100 |
| train batch size | 1600 |
| learning rate | 5e-5 |
| # fully connected layers | (64, 64) |
| # post FCNet hidden layers | 256 |
| LSTM size | 256 |
| SGD minibatch size | 128 |
| # SGD iteration | 30 |
| GAE lambda | 1 |
| KL coefficient | 0.2 |
| KL target | 0.01 |
| clip parameter | 0.3 |
| value function clip parameter | 10 |

Table 8: Hyperparameters for RLlib PPO policies.

C Additional Results

This section presents additional results and statistical analyses to complement the main results.

C.1 Fixed-Bobs Evaluation

A bar plot version of the Figure 2 bottom row is shown in Figure 4. The numerical values are identical with additional error bars corresponding to 95% confidence intervals for better comparison. Conclusions are the same as mentioned in Section 7.1.

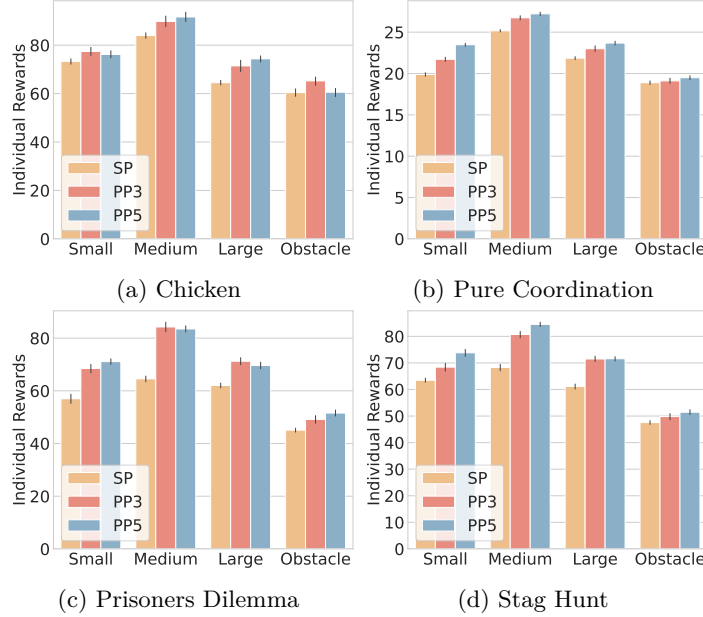


Figure 4: Fixed-Bobs evaluation of agents trained by self-play (SP), population-play $p=3$ (PP3), population-play $p=5$ (PP5) across four scenarios. Error bars correspond to 95% confidence intervals calculated over all populations and seeds (5 seeds for SP) with 10 independent games between each Alice-Bob combination. *Result:* With a growing population, PP gains a larger advantage over SP in general. However, percentage increments vary across different scenarios under the same environment, and the overall trends of improvement vary across different environments.

C.2 Resource Allocation

In this section, we demonstrate the effect of Bob’s population number on LoI approximation and resource allocation. Following Section 4.2 we estimate LoI by training $a+b$ SP policies with $a=1$ and $b \in \{1, 2, 3, 4\}$. For each b , we train 5 independent sets of SP policies (*i.e.*, 5 different sets of $a+b$ SP policies) and compute the LoIs for each set, respectively. Subsequently, We compute the variance of the estimated LoIs and report the results in Figure 5.

We can observe that augmenting Bob’s population size leads to a notable reduction in the variance of LoI estimation. This trend persists across all scenarios and environments. However, this reduction comes at the expense of additional computational resources.

Next, we proceed with the resource allocation test outlined in Section 6.2, comparing the LoI-guided heuristic allocation and uniform allocation using the previously calculated LoI values. We define *mean advantage* as the average normalized individual rewards’ disparity between the heuristic and uniform cases across 5 independent experiments. The results are reported in Figure 6.

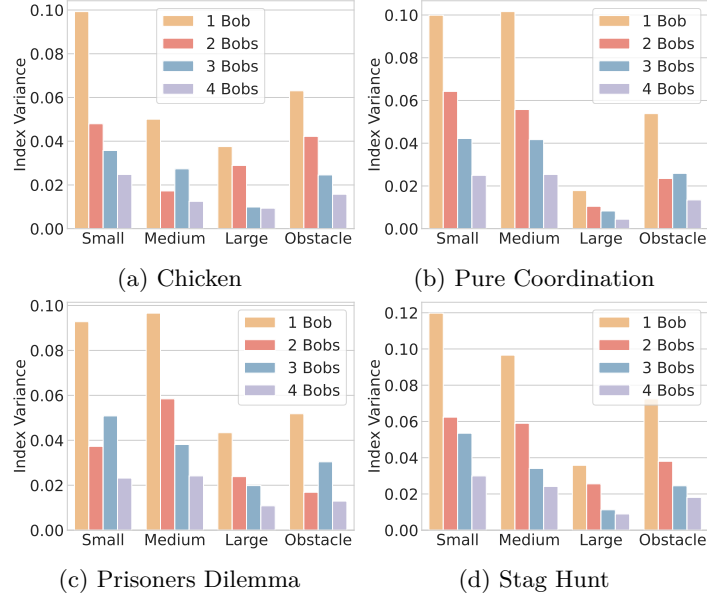


Figure 5: Effect of Bob's population size b on the variance of LoI approximation. Variance is calculated on 5 individual tests. *Result:* Increasing Bob's population size lowers the LoI estimation variance across all scenarios and environments.

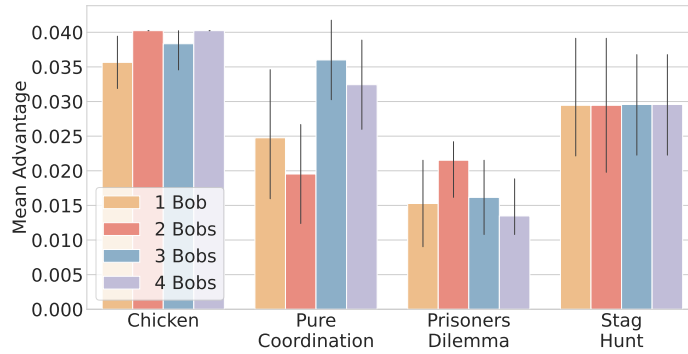


Figure 6: Effect of Bob's population size b on the advantage of LoI-guided heuristic allocation over uniform allocation. Error bars correspond to **95%** confidence intervals calculated over 5 independent comparisons. *Result:* The increase in Bob's population size does not impact the advantage in the average performance of LoI-guided heuristic allocation over uniform allocation.

It is evident that there is not a distinct correlation between Bob's population size used in LoI calculation and the advantage of LoI-guided heuristic allocation over uniform allocation. (*i.e.*, lower variance in LoI approximation does not necessarily impact the performance of heuristic allocation based on that index). As a result, smaller population size for LoI estimation can still attain comparable performance in the resource allocation task using the proposed heuristic method.

D Methods

In this section, we present the pseudocode for LoI-guided resource allocation implementation (Algorithm 2).

Algorithm 2 LoI-guided resource allocation

Input: # scenarios n , LoIs for given scenarios I

- 1: Get mean LoI across scenarios $\bar{I} \leftarrow \text{MEAN}(I)$
- 2: $\sigma \leftarrow \text{STD}(I)$
- 3: $I_l \leftarrow \bar{I} - \sigma$
- 4: $I_u \leftarrow \bar{I} + \sigma$
- 5: Initialize set of training method $\mathcal{G} \leftarrow \{\}$
- 6: Initialize count $c \leftarrow 0$
- 7: **for** $i=1:n$ **do**
- 8: **if** $I_i < I_l$ **then**
- 9: $\mathcal{G} \leftarrow \mathcal{G} \cup \text{'SP'}$
- 10: $c \leftarrow c + 1$
- 11: **else if** $I_i > I_u$ **then**
- 12: $\mathcal{G} \leftarrow \mathcal{G} \cup \text{'PP5'}$
- 13: $c \leftarrow c - 1$
- 14: **else**
- 15: $\mathcal{G} \leftarrow \mathcal{G} \cup \text{'PP3'}$
- 16: **end if**
- 17: **end for**
- 18: **if** $c \neq 0$ **then** \triangleright Adjust methods to keep the total budget
- 19: **if** $c > 0$ **then** \triangleright Need more 'PP5'
- 20: $k \leftarrow \text{ARGMAX}(\mathcal{G})$
- 21: $\mathcal{G}_k \leftarrow \text{'PP5'}$
- 22: **else** \triangleright Need more 'SP'
- 23: $k \leftarrow \text{ARGMIN}(\mathcal{G})$
- 24: $\mathcal{G}_k \leftarrow \text{'SP'}$
- 25: **end if**
- 26: **end if**

Output: \mathcal{G}
