Supplemental Material for "Improved Response Strategies for Unknown Behaviours in Multiagent Interaction"

Yinghui Pan (Member, IEEE), Mengen Zhou, Yifeng Zeng (Member, IEEE), Biyang Ma (Member, IEEE), Yew-soon Ong (Fellow, IEEE) and Guoquan Liu

In this Supplementary Materials, we detail the frameworks of PSB and ACB, PSB operators, and shared functionalities. All implementations are included in our I-DID toolkit [1] .

A: PSB and ACB Implementation

We introduce PSB and ACB frameworks in Figs. 1 and 2, respectively, and elaborate on PSB operators and their shared functionalities with ACB.

A.1: [The framework of PSB and ACB]

Fig. 1 shows our PSB (Particle Swarm Optimization-enabled Behavior) framework. First, known policy trees become particles $\sigma = (p, v)$ forming pop (①). Fitness $F(\sigma)$ is calculated (②) to find local best $\bar{\sigma}$ and global best σ^* (③). Over N iterations, positions and velocities update (④-⑤), fitness recalculates (⑥), and best particles update (⑦). Particles move towards σ^* or $\bar{\sigma}$ (⑧). Finally, particles transform back to policy trees and decode into behaviors via $transform$ (⑨), selecting top-K based on fitness (⑩). Behaviors, initialized with varied beliefs, optimize over iterations.

Fig. 2 presents our ACB (Ant Colony Optimization-enabled Behavior) framework. Ant populations and pheromone tables initialize (①-②). Pheromones update based on ant actions (③-④). Ants choose next actions epsilon-greedily, influenced by pheromones (⑤). After N iterations (⑥), the population converts to policy trees, selecting top-K behaviors (⑦-⑧).

A.2: [PSB Operators]

We introduce four PSB operators:

- *Minus* ($\ominus$): Computes particle velocity from positions.
- *Plus* ($\oplus$): Calculates new particle positions.
- *Times* ($\otimes$): Scales velocity v_1 by scalar ω.
- *Merging* ($\uplus$): Combines velocities v_1 and v_2.

A.3: [The Common Operators in PSB and ACB]

Both PSB and ACB algorithms utilize four fundamental operators detailed in Alg. 2. The operators in PSB and ACB enable transformations between policy trees and particles/ants.

- Formalise: Generates an action sequence from a policy tree.

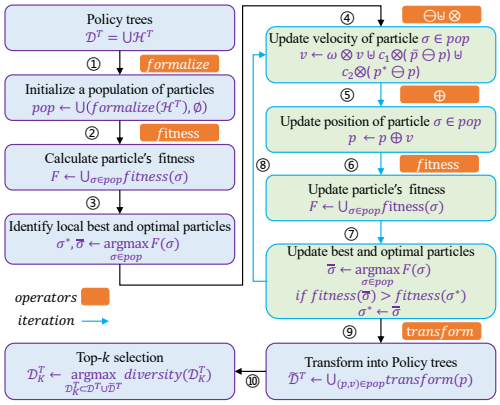

Fig. 1: PSO-enabled behavior generation framework comprises main components: initialization and updates of particles' positions and velocities.

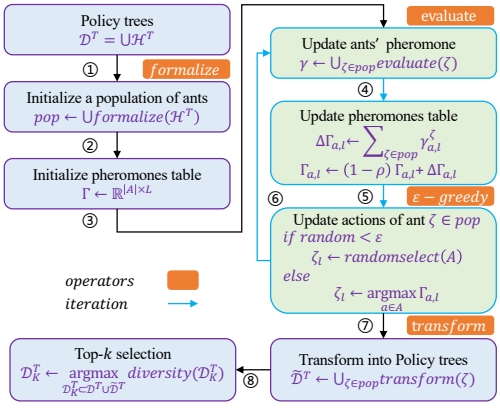

Fig. 2: The ACB algorithm generates new behaviors by updating the pheromone table, maintaining an ant population where each ant performs action sequences, updated upon selecting new actions.

- Transform: Converts action sequences to policy trees.
- Fitness: Evaluates particle sequences using GeNIe decision tool [2].
- Evaluate: Calculates pheromone for ant policies.

[1] https://github.com/lamingic/SI-IDID

[2] https://www.bayesfusion.com

Algorithm 1: The Four Operators in PSB

Function $\ominus(\ p_1, p_2)$ $\quad \triangleleft\ v \leftarrow p_1 \ominus p_2$**:**
  $v \leftarrow \emptyset$
  for $l \in \{1, 2, \cdots |p_1|\}$ **do**
    if $p_1[l] \neq p_2[l]$ **then**
      $v \leftarrow v \bigcup (l, p_1[l], 1/|A|)$
    end
  end
return v
Function $\oplus(\ p_1, v)$ $\quad \triangleleft\ p \leftarrow p_1 \oplus v$**:**
  $p \leftarrow p_1$
  for $(l, a, P) \in v$ **do**
    if $random() > P$ **then**
      $a \leftarrow randomselect(A)$
    end
    $p[l] \leftarrow a$
  end
return p
Function $\otimes(\ v_1, \omega)$ $\quad \triangleleft\ v \leftarrow \omega \otimes v_1$**:**
  $v \leftarrow \emptyset$
  for $(l, a, P) \in v_1$ **do**
    $v \leftarrow v \bigcup (l, a, min(\omega * P, 1))$
  end
return v
Function $\uplus(\ v_1, v_2)$ $\quad \triangleleft\ v \leftarrow v_1 \uplus v_2$**:**
  $v \leftarrow \emptyset$
  for $(l, a, P) \in v_1$ **do**
    for $(l', a', P') \in v_2$ **do**
      if $l == l'$ & $a == a'$ **then**
        $v \leftarrow v \bigcup (l', a', min(P + P', 1))$
      end
      else
        $v \leftarrow v \bigcup (l, a, P)$
        $v \leftarrow v \bigcup (l', a', P')$
      end
    end
  end
return v

Algorithm 2: The operators in PSB and ACB

Function $formalise(\ \mathcal{H}^T)$**:**
  $\triangleleft$ *generate action sequence of tree* $\mathcal{H}^T$
  $p \leftarrow \{a_l | a_l \leftarrow 0, \forall l \in \{1, 2, \cdots \frac{|\Omega|^T - 1}{|\Omega| - 1}\}\}$
  for $(h^T, i) \in \mathcal{H}^T$ **do**
    $\{a_1, o_2, \cdots, o_T, a_T\} \leftarrow h^T$
    for $t \in \{1, 2, \cdots T\}$ **do**
      $j \leftarrow \frac{|\Omega|^{(t-1)} - 1}{|\Omega| - 1} + \lfloor \frac{i-1}{|\Omega|^{(t-1)}} \rfloor + 1$
      $p[j] \leftarrow a_t$
    end
  end
return p
Function $transform(p)$**:**
  $\triangleleft$ *generate policy tree from action sequence* p
  $\mathcal{H}^T \leftarrow \bigcup h^T$
  for $(h^T, i) \in \mathcal{H}^T$ **do**
    $\{a_1, o_2, \cdots, o_T, a_T\} \leftarrow h^T$
    for $t \in \{1, 2, \cdots T\}$ **do**
      $j \leftarrow \frac{|\Omega|^{(t-1)} - 1}{|\Omega| - 1} + \lfloor \frac{i-1}{|\Omega|^{(t-1)}} \rfloor + 1$
      $a_t \leftarrow p[j]$
      if $t \neq T$ **then**
        $k \leftarrow \lfloor \frac{(i-1)|\Omega|^t}{|\Omega|^{(T-1)}} \rfloor + 1$
        $o_{t+1} \leftarrow o^k$
      end
    end
    $h^T \leftarrow \{a_1, o_2, \cdots, o_T, a_T\}$
  end
return $\mathcal{H}^T$
Function $fitness(\sigma)$**:**
  $\triangleleft$ *evaluate the fitness of action sequence* σ
  $\mathcal{H}^T \leftarrow transform(\sigma)$
  for $s_i \in S$ **do**
    $V_{\mathcal{H}^T}(s_i) \leftarrow R(s, a(\mathcal{H}^T)) + \lambda \sum_{\forall s' \in S} Pr_{a(\mathcal{H}^T)}(s'|s)[\sum_{\forall o \in O} Pr_{a(\mathcal{H}^T)}(o|s') V_{o(\mathcal{H}^T)}(s')]$
  end
  $\alpha_{\mathcal{H}^T} \leftarrow [V_{\mathcal{H}^T}(s_1), V_{\mathcal{H}^T}(s_2), \cdots, V_{\mathcal{H}^T}(s_{|S|})]$
  $f \leftarrow \sum_{\forall s \in S} b^0(s) \alpha_{\mathcal{H}^T}(s)$
return f
Function $evaluate(\zeta)$**:**
  $\triangleleft$ *evaluate the expected reward of action sequence* σ
  $\mathcal{H}^T \leftarrow transform(\zeta)$
  $ER \leftarrow \mathbb{R}^{|A| \times \frac{|\Omega|^T - 1}{|\Omega| - 1}}$
  for $(h^T, i) \in \mathcal{H}^T$ **do**
    $\{a_1, o_2, \cdots, o_T, a_T\} \leftarrow h^T$
    for $t \in \{1, 2, \cdots T\}$ **do**
      $l \leftarrow \frac{|\Omega|^{(t-1)} - 1}{|\Omega| - 1} + \lfloor \frac{i}{|\Omega|^{(t-1)}} \rfloor$
      for $s \in S$ **do**
        $b^t(s) \leftarrow \frac{Pr(o_l|s, a_l \sum_{s' \in S} Pr(s|s', a_l) Pr(s'|b^{t-1})}{\sum_{s' s'' \in S} Pr(o_l|s'', a_l) Pr(s''|s', a_l) Pr(s'|b^{t-1})}$
      end
      $ER(a_l, l) \leftarrow \sum_{s \in S} b^t(s) R(s, a_l)$
    end
  end
return ER