Supplementary Materials for "Non-Cooperative Inverse Reinforcement Learning"

A Illustrative Example

Consider a simple zero-sum patrolling game where a thief A aims to steal valuables from a museum m and a gallery g that a security guard D is watching over. The state space is defined as the set of possible locations of the players (A, D), defined as $\mathcal{S} = \{(m,m), (m,g), (g,m), (g,g)\}$. The initial state is uniformly sampled from $\mathcal{S}$. The game is played over multiple stages, where at each stage of the game, A and D observe the current state $s \in \mathcal{S}$ and (simultaneously) choose to either stay at their respective locations or switch to the other location, *i.e.*, $\mathcal{A} = \mathcal{D} = \{\text{stay}, \text{switch}\}$. The state deterministically transitions from s to s' based on (a, d). If the state transitions to either $s' = (m, g)$ or $s' = (g, m)$, A will successfully steal an item with probability 1 and gain a reward of either θ if $s' = (m, g)$, or $1 - \theta$ if $s' = (g, m)$, where $\theta \in [0, 1]$ reflects A's private preference that is unknown to D. If the state transitions to either $s' = (m, m)$ or $s' = (g, g)$, the presence of D will lower A's probability of success to $1/2$. Thus, A's (expected) reward at each stage is

$$R(s;\theta) = \begin{cases} \theta/2 & \text{if } s = (m,m) \\ \theta & \text{if } s = (m,g) \\ 1-\theta & \text{if } s = (g,m) \\ (1-\theta)/2 & \text{if } s = (g,g) \end{cases} \tag{8}$$

Under this setting, we compare the strategies obtained under the MA-IRL formalism, *e.g.*, [21, 41, 20], to the strategies where D can learn the intent adaptively under the N-CIRL formalism. In MA-IRL, D has access to an attack log, which reflects that A prefers the gallery twice as much as the museum, *i.e.*, $\theta = 1/3$. In an initial *learning phase*, D learns from previous equilibrium behavior[5], and in the subsequent *execution phase*, A and D play the game described above. In contrast, D under N-CIRL has no prior knowledge of A's intent and learns it from scratch through information that is revealed as the game unfolds. For purposes of this example, A's preferences are assumed to flip in the execution phase, *i.e.*, $\theta = 2/3$. We now compare two strategies in a two-stage instance of the above game.

MA-IRL Strategies. For $\theta = 1/3$, D has a unique pure Nash equilibrium strategy that generates its next location to be g, independent of the initial state. However, since A's true intent in the execution phase when D actually participates is $\theta = 2/3$, A's pure Nash equilibrium strategies is to go to m. Such a combination of A and D's equilibrium strategies yields a next state of (m, g), which results in a reward of $2/3$ for A. Hence, over the two-stage game (execution phase), the total reward for A against D under MA-IRL is $2 \cdot 2/3 = 4/3$.

N-CIRL Strategies. D has a uniform patrol strategy in the first stage (toss a fair coin to decide where to patrol). Since A has intent parameter $\theta = 2/3$ in the execution phase, it will go to the museum according to $R(s; \theta = 2/3)$, with state (m, m) arising from the Nash equilibrium strategies. Hence, A's expected reward in the first stage is $1/2 \cdot 1/2 \cdot 2/3 + 1/2 \cdot 2/3 = 1/2$. At the second stage, D receives the observation that A went to m, thus infers $\theta > 1/2$. With the new estimation of A's intent, D prefers to defend m, which results in a cost of $-1/2 \cdot \theta$ that is smaller than the cost of defending g, which is $-\theta$. Therefore, the expected reward of A at the second stage becomes $1/2 \cdot 2/3 = 1/3$, and the total reward in the two-stage game is $1/2 + 1/3 = 5/6 < 4/3$.

The above example illustrates that there exist settings where the defender can incur a lower cost by interleaving learning and execution.

---

[5] D under MA-IRL is (generously) assumed to be able to *perfectly recover* the parameter $\theta = 1/3$ during the learning phase. This is often not the case as D's inference using historical data can not be exactly accurate in practice. We make this assumption to favor MA-IRL as much as possible.

B Proofs of Main Results

B.1 Proof of Lemma 1

Proof. Given a current state s and distribution b on Θ, the posterior distribution is computed by conditioning on the new information (consisting of the actions (a, d) and updated state s') as

$$
\begin{aligned}
P(\theta = \vartheta \mid s, b, a, d, s') &= \frac{P(s, b, a, d, s' \mid \vartheta)P(\theta = \vartheta)}{\sum_{\vartheta'} P(s, b, a, d, s' \mid \vartheta')P(\theta = \vartheta')} && (9)\\
&= \frac{P(s \mid \vartheta)P(a, d \mid s, \vartheta)P(b, s' \mid s, a, d, \vartheta)P(\theta = \vartheta)}{\sum_{\vartheta'} P(s, b, a, d, s' \mid \vartheta')P(\theta = \vartheta')} && (10)\\
&= \frac{P(s \mid \vartheta)P(a \mid s, \vartheta)P(d \mid s)P(s' \mid s, a, d)P(b \mid s, \vartheta)P(\theta = \vartheta)}{\sum_{\vartheta'} P(s, b, a, d, s' \mid \vartheta')P(\theta = \vartheta')} && (11)\\
&= \frac{P(s \mid \vartheta)P(a \mid s, \vartheta)P(d \mid s)P(s' \mid s, a, d)P(b \mid s, \vartheta)b(\vartheta)}{\sum_{\vartheta'} P(s \mid \vartheta')P(a \mid s, \vartheta')P(d \mid s)P(s' \mid s, a, d)P(b \mid s, \vartheta')b(\vartheta')} && (12)\\
&= \frac{P(s' \mid s, a, d)P(d \mid s)P(a \mid s, \vartheta)b(\vartheta)}{P(s' \mid s, a, d)P(d \mid s)\sum_{\vartheta'} P(a \mid s, \vartheta')b(\vartheta')} && (13)\\
&= \frac{P(a \mid s, \vartheta)b(\vartheta)}{\sum_{\vartheta'} P(a \mid s, \vartheta')b(\vartheta')}, && (14)
\end{aligned}
$$

where Eq. (9) follows Bayes rule, Eq. (10) uses definition of conditional probability, Eq. (11) uses the conditional independence of a and d given s, ϑ, and that of b and s' given s, a, d, ϑ, Eq. (12) expands the denominator as in Eq. (11) and uses the fact that $P(\theta = \vartheta') = b(\vartheta')$, Eq. (13) uses that $P(s \mid \vartheta) = P(b \mid s, \vartheta) = 1$, and Eq. (14) follows by cancelling out $P(s' \mid s, a, d)$. Note that the probabilities $P(s' \mid s, a, d), P(d \mid s), P(a \mid s, \vartheta)$ are all nonzero, as the calculation in Eqs. (9)-(14) is only for those tuples of (a, d, s') that have been realized. Recognizing that $P(a \mid s, \vartheta) = \bar{\pi}^A(a \mid s, \vartheta)$ yields the result. ☐

B.2 Proofs of Propositions 1 and 2

Proof. The proofs of Propositions 1 and 2 are similar and are built upon the results of [35]. In the language of [35], player 1 is the more informed player and player 2 is the less informed player. Let I and J denote the action sets of player 1 and 2, respectively, K denotes the set of the states of the world, X denotes the set of the stochastic states, and $\lambda \in (0, 1)$ is the discount factor. Let $s \in \Delta(I)^K$ denote the strategy[6] of player 1 and $t \in \Delta(J)$ denote the strategy of player 2. To prove Proposition 1, consider the following sequential decomposition from [35, Proposition 6],

$$
\begin{aligned}
v_\lambda(p, x) = \max_{s \in \Delta(I)^K} \min_{t \in \Delta(J)} \Bigg[&\lambda \sum_{k \in K} \sum_{(i,j) \in I \times J} p^k s^k(i) A^{kx}_{ij} t(j) \\
&+ (1 - \lambda) \sum_{k \in K} \sum_{(i,j) \in I \times J} \sum_{y \in X} p^k s^k(i) t(j) q(x, i, j, y) v_\lambda(p_i, y) \Bigg], \qquad (15)
\end{aligned}
$$

where superscripts denote indexing and p_i denotes the Bayesian update defined elementwise by $p_i^k = \frac{p^k s^k(i)}{\sum_{l \in K} p^l s^l(i)}$. There are two modifications that need to be made to the recursive formula. First, compared to the reward function A in [35], the reward function in N-CIRL additionally depends on the successor state s'. This dependence requires that the first term in the summation of Eq. (15) also needs to take an expectation over s'. Second, the form of the payoff differs in N-CIRL; dividing Eq. (15) by λ yields an expression for v_λ / λ which corresponds to value of the primal game v in Proposition 1. Denoting $s, t, p, A, q, 1 - \lambda$ therein by $\bar{\pi}^A, \bar{\pi}^D, b, R, \mathcal{T}, \gamma$ in our notation system, respectively, yields the sequential decomposition of (2) in Proposition 1.

---

[6] The notation $\Delta(I)^K$ means all functions from K to $\Delta(I)$.

To prove Proposition 2, consider the sequential decomposition from [35, Proposition 7],

$$\begin{aligned} w_\lambda(\alpha, x) = &\inf_{t\in\Delta(J)} \inf_{\beta\in\mathbb{R}^{K\times I\times X}} \sup_{\pi\in\Delta(I\times K)} \sum_{i\in I}\sum_{k\in K} \pi(i,k)\Big(\lambda t(j) A_{ij}^{kx} + \alpha^k\Big) \\ &- \sum_{k\in K}\sum_{(i,j)\in I\times J}\sum_{y\in X} (1-\lambda)\pi(i,k)\beta^k(i,y)t(j)q(x,i,j,y) \\ &+ \sum_{i\in I}\sum_{y\in X}(1-\lambda)\Big(\sum_{j\in J} t(j)q(x,i,j,y)\Big)\pi(i)w_\lambda(\beta(i,y),y) \end{aligned} \tag{16}$$

where $\pi(i) = \sum_{k\in K}\pi(i,k)$ and $\alpha \in \mathbb{R}^K$. First, note that the dual game in our case is finite (I, J, K, X are finite in our problem) and hence has a value. As a result, the $\inf$ and $\sup$ in [35, Propositions 7] can be replaced by $\min$ and $\max$. Next, as in the proof of Proposition 1, divide Eq. (16) by λ and denote $t, \alpha/\lambda, \beta/\lambda, \pi, \omega_\lambda/\lambda, 1-\lambda$ therein by $\bar{\pi}^D, \zeta, \xi, \mu, w, \gamma$, respectively. Grouping terms, we arrive at the sequential decomposition of (5) in Proposition 2. ☐

B.3 Proof of Lemma 2

Proof. We prove this lemma by showing that the value backup operator G for a given one-stage strategy profile $(\bar{\pi}^A, \bar{\pi}^D)$ is a contraction mapping with $\gamma \in [0,1)$. Let $\eta := (s,b)$ and v, v' be value functions. By Blackwell's sufficiency theorem [5], $[Gv](\eta)$ is a contraction mapping if it satisfies i) monotonicity: $[Gv](\eta) \geq [Gv'](\eta)$, if $v(\eta) \geq v'(\eta)$ for any η, and ii) discounting: $[Gv](\eta) = [Gv'](\eta) + \gamma\varepsilon$, $\forall v(\eta) = v'(\eta) + \varepsilon$. To show monotonicity, assume $v(\eta) \geq v'(\eta)$ for all η; then we have

$$\sum_{a,d,s',\vartheta} b(\vartheta)\bar{\pi}^A(a \mid s,\vartheta)\bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)(v(\eta') - v'(\eta')) \geq 0 \quad \forall \eta$$

where η' represents the updated attacker information state. Note that the instantaneous reward does not depend on v, v', thus

$$\begin{aligned} [Gv](\eta) - [Gv'](\eta) &= \gamma \max_{\bar{\pi}^A}\min_{\bar{\pi}^D}\left\{V_{\bar{\pi}^A,\bar{\pi}^D}(v;\eta) - V_{\bar{\pi}^A,\bar{\pi}^D}(v';\eta)\right\} \\ &= \gamma \max_{\bar{\pi}^A}\min_{\bar{\pi}^D}\Big\{\sum_{a,d,s',\vartheta} b(\vartheta)\bar{\pi}^A(a \mid s,\vartheta)\bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)(v(\eta') - v'(\eta'))\Big\} \\ &\geq 0. \end{aligned}$$

To show discounting, let $v(\eta) = v'(\eta) + \varepsilon$. Then

$$\begin{aligned} [Gv](\eta) &= \max_{\bar{\pi}^A}\min_{\bar{\pi}^D}\Bigg\{\sum_{a,d,s',\vartheta} b(\vartheta)\bar{\pi}^A(a \mid s,\vartheta)\bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)R(s,a,d,s';\vartheta) \\ &\qquad + \gamma\sum_{a,d,s',\vartheta} b(\vartheta)\bar{\pi}^A(a \mid s,\vartheta)\bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)(v'(\eta') + \varepsilon)\Bigg\} \\ &= \max_{\bar{\pi}^A}\min_{\bar{\pi}^D}\Bigg\{\sum_{a,d,s',\vartheta} b(\vartheta)\bar{\pi}^A(a \mid s,\vartheta)\bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)R(s,a,d,s';\vartheta) \\ &\qquad + \gamma\sum_{a,d,s',\vartheta} b(\vartheta)\bar{\pi}^A(a \mid s,\vartheta)\bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)(v'(\eta'))\Bigg\} + \gamma\varepsilon \\ &= [Gv'](\eta) + \gamma\varepsilon. \end{aligned}$$

Therefore, the one-stage value backup operator is a contraction mapping for a given one-stage strategy profile $(\bar{\pi}^A, \bar{\pi}^D)$. ☐

B.4 Proof of Lemma 3

Proof. Let w, w' be value functions. As in the proof of Lemma 2, we show the monotonicity and discounting properties, and then invoke Blackwell's sufficiency theorem. Assume $w(\xi, s) \geq w'(\xi, s)$

for all s, ξ; then

$$\sum_{a,d,s',\vartheta} \mu(a,\vartheta)\bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)(w(\xi_{a,s'},s') - w'(\xi_{a,s'},s')) \geq 0 \quad \forall s,\xi$$

Note that the instantaneous reward does not depend on w, w', and thus

$$\begin{aligned}
&[Hw](s,\zeta) - [Hw'](s,\zeta) \\
&= \gamma \min_{\bar{\pi}^D,\xi} \max_{\mu} \big\{ W_{\bar{\pi}^D,\mu}(w,\xi;s) - W_{\bar{\pi}^D,\mu}(w',\xi;s) \big\} \\
&= \gamma \min_{\bar{\pi}^D,\xi} \max_{\mu} \big\{ \sum_{a,d,s',\vartheta} \mu(a,\vartheta)\bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)(w(\xi_{a,s'},s') - w'(\xi_{a,s'},s')) \big\} \\
&\geq 0.
\end{aligned}$$

To show discounting, let $w(\xi,s) = w'(\xi,s) + \varepsilon$ for all s,ξ. Then

$$\begin{aligned}
[Hw](s,\zeta) &= \min_{\bar{\pi}^D,\xi} \max_{\mu} \Bigg\{ \sum_{a,\vartheta} \mu(a,\vartheta)\Big(\zeta(\vartheta) + \sum_{d,s'} \bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)\mathcal{R}(s,a,d,s';\vartheta) \Big) \\
&\qquad + \gamma \sum_{a,d,s',\vartheta} \mu(a,\vartheta)\bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)\big(w(\xi_{a,s'},s') - \xi_{a,s'}(\vartheta)\big) \Bigg\} \\
&= \min_{\bar{\pi}^D,\xi} \max_{\mu} \Bigg\{ \sum_{a,\vartheta} \mu(a,\vartheta)\Big(\zeta(\vartheta) + \sum_{d,s'} \bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)\mathcal{R}(s,a,d,s';\vartheta) \Big) \\
&\qquad + \gamma \sum_{a,d,s',\vartheta} \mu(a,\vartheta)\bar{\pi}^D(d \mid s)\mathcal{T}(s' \mid s,a,d)\big(w'(\xi_{a,s'},s') - \xi_{a,s'}(\vartheta)\big) \Bigg\} + \gamma\varepsilon \\
&= [Hw'](s,\zeta) + \gamma\varepsilon.
\end{aligned}$$

Therefore, the one-stage value backup operator is a contraction mapping for a given $(\bar{\pi}^D, \xi, \mu)$. □

B.5 Proof of Lemma 4

Proof. By definition of the subroutine SAWTOOTH-A in Algorithm 1, for given $\mathcal{Y}_s, \mathcal{W}_s$, the function $\Upsilon_v(\mathcal{Y}_s, \mathcal{W}_s, \cdot)$ returns the x_j for the j that makes $v_j - c^T b_j > 0$. Denote this j by j^*. Let $e_{j,\vartheta} = \big(0, \cdots, 1/b_j(\vartheta), \cdots, 0\big)^T \in \mathbb{R}^{|\Theta|}$ be an all-zero vector except that the ϑ-th element is $1/b_j(\vartheta)$. Thus, the following equivalence relationship holds, *i.e.,* for any V

$$V \leq \Upsilon_v\big(\mathcal{Y}_s, \mathcal{W}_s, b\big) \iff V \leq c^T b + \min_{\vartheta \in \Theta} \big\{ e^T_{j^*,\vartheta} b | b_{j^*}(\vartheta) > 0 \big\} \cdot (v_{j^*} - c^T b_{j^*}). \tag{17}$$

The positivity of $v_{j^*} - c^T b_{j^*}$ implies that (17) can then equivalently be written as

$$V \leq \Upsilon_v\big(\mathcal{Y}_s, \mathcal{W}_s, b\big) \iff V \leq c^T b + e^T_{j^*,\vartheta} b \cdot (v_{j^*} - c^T b_{j^*}), \quad \forall \vartheta \in \Theta, \tag{18}$$

which essentially describes $|\Theta|$ constraints that are linear in b. Note that the dependences of the constraints (18) on $\mathcal{W}_s$ and $\mathcal{Y}_s$ are implicitly embedded in finding c and (b_{j^*}, v_{j^*}), respectively.

Similarly, SAWTOOTH-D in Algorithm 1 returns the y_j that makes $w_j - c^T \zeta_j < 0$. Denoting this j by j^*, we have the following equivalent conditions

$$V \geq \Upsilon_w\big(\mathcal{Y}_s, \mathcal{W}_s, \zeta\big) \iff V \geq c^T \zeta + d^T_{j^*,\vartheta} b \cdot (w_j - c^T \zeta_j), \quad \forall \vartheta \in \Theta, \tag{19}$$

where $d_{j^*,\vartheta} = \big(0, \cdots, 1/\zeta_j(\vartheta), \cdots, 0\big)^T \in \mathbb{R}^{|\Theta|}$ be an all-zero vector except that the ϑ-th element is $1/\zeta_j(\vartheta)$. Note that (19) describes $|\Theta|$ constraints that are linear in ζ. This completes the proof. □

C Details of the NC-PBVI Algorithm

The experimental setup is as follows. We randomly generate attack graphs with sizes ranging from 6 to 10 nodes. Root nodes are assumed to be enabled initially. Furthermore, we limit the in-degree and out-degree of nodes to be at most 3. For each graph of size n, we run an experiment on a finite horizon of length n. The intent parameter is uniformly chosen from a set of random intent parameters of size $|\Theta| = 10$. The attacker's accumulated reward is collected at each stage and normalized by the total reward across all nodes. To compare the average performance of N-CIRL and MA-IRL, we run 20 graph instances for each size and plot the attackers' average reward, see Figure 1. Note that in MA-IRL, both players are playing a complete information game. The difference is that the defender is playing Nash equilibrium strategies based on an intent parameter that is inferred from existing attack data. The attacker, on the other hand, knows its true intent (which is in general different from the defender's inferred intent) and plays its corresponding Nash equilibrium strategies.

All the experiments were run on a machine with an AMD Ryzen 1950X Processor and 32GB of RAM. We used GUROBI 8.1.1 to solve the LPs used in our algorithm and the probability of success, β_{iy}, is assumed to be 0.8. The detailed pseudocode of the proposed NC-PBVI algorithm is summarized in Algorithm 1.

Algorithm 1 Non-Cooperative Point-Based Value Iteration (NC-PBVI)

```
function NC-PBVI ($b_0, \zeta_0, N, T$)
    for $p$ in $\{A, D\}$ do
        for $s \in \mathcal{S}$ do
            Initialize $\mathcal{Y}_s^p, \mathcal{W}_s^p$
        end for
        for N expansions do
            for T iterations do
                $\mathcal{Y}^p, \mathcal{W}^p \leftarrow$ UPDATE-P($\mathcal{Y}^p, \mathcal{W}^p$)
            end for
            $\mathcal{Y}^p, \mathcal{W}^p \leftarrow$ EXPAND-P ($\mathcal{Y}^p, \mathcal{W}^p$)
        end for
    end for
    Compute $\pi^A, \pi^D$ by solving $P_A(s, b)$ and
    $P_D(s, \zeta)$ for all $s$ and finite sets of $b$ (resp. $\zeta$).
end function

function UPDATE-A ($\mathcal{Y}, \mathcal{W}$)
    for $s \in \mathcal{S}$ do
        for $(b, v) \in \mathcal{Y}_s \cup \mathcal{W}_s$ do
            update $v$ by solving $P_A(s, b)$
        end for
    end for
    return $\mathcal{Y}, \mathcal{W}$
end function

function UPDATE-D ($\mathcal{Y}, \mathcal{W}$)
    for $s \in \mathcal{S}$ do
        for $(\zeta, w) \in \mathcal{Y}_s \cup \mathcal{W}_s$ do
            update $w$ by solving $P_D(s, \zeta)$
        end for
    end for
    return $\mathcal{Y}, \mathcal{W}$
end function

function EXPAND-A ($\mathcal{Y}, \mathcal{W}$)
    for $s \in \mathcal{S}$ do
        for $(b, v) \in \mathcal{Y}_s \cup \mathcal{W}_s$ do
            $\Omega \leftarrow \varnothing$
            $\bar{\pi}^A \leftarrow$ solve $P_A(s, b)$
            for $a \in \mathcal{A}(s)$ do
                $b_a \leftarrow \tau(s, b, a)$
                $\Omega \leftarrow \Omega \cup b_a$
            end for
            $b' \leftarrow \arg\max_{b_a \in \Omega} \sum_{\vartheta \in \Theta} |b_a(\vartheta) - b(\vartheta)|$
            if $(b', \cdot) \notin \mathcal{Y}_s \cup \mathcal{W}_s$ then
                $V_{s,b'} \leftarrow$ solve $P_A(s, b')$
                $\mathcal{Y}_s \leftarrow \mathcal{Y}_s \cup (b', V_{s,b'})$
            end if
        end for
    end for
    return $\mathcal{Y}, \mathcal{W}$
end function

function EXPAND-D ($\mathcal{Y}, \mathcal{W}$)
    for $s \in \mathcal{S}$ do
        for $(\zeta, w) \in \mathcal{Y}_s \cup \mathcal{W}_s$ do
            $\zeta' \leftarrow$ solve $P_D(s, \zeta)$
            if $(\zeta', \cdot) \notin \mathcal{Y}_s \cup \mathcal{W}_s$ then
                $W_{s,\zeta'} \leftarrow$ solve $P_D(s, \zeta')$
                $\mathcal{Y}_s \leftarrow \mathcal{Y}_s \cup (\zeta', W_{s,\zeta'})$
            end if
        end for
    end for
    return $\mathcal{Y}, \mathcal{W}$
end function

function SAWTOOTH-A ($\mathcal{Y}_s, \mathcal{W}_s, b$)
    for $(b_i, v_i) \in \mathcal{W}_s$ do
        $c_i \leftarrow v_i$
    end for
    $x_j \leftarrow c^T b$
    for $(b_j, v_j) \in \mathcal{Y}_s$ do
        if $v_j - c^T b_j > 0$ then
            $\phi \leftarrow \min_{\vartheta \in \Theta} \{b(\vartheta)/b_j(\vartheta) | b_j(\vartheta) > 0\}$
            $x_j \leftarrow x_j + \phi(v_j - c^T b_j)$
            break
        end if
    end for
    return $x_j$
end function

function SAWTOOTH-D ($\mathcal{Y}_s, \mathcal{W}_s, \zeta$)
    for $(\zeta_i, v_i) \in \mathcal{W}_s$ do
        $c_i \leftarrow v_i$
    end for
    $y_j \leftarrow c^T \zeta$
    for $(\zeta_j, w_j) \in \mathcal{Y}_s$ do
        if $w_j - c^T \zeta_j < 0$ then
            $\psi \leftarrow \min_{\vartheta \in \Theta} \{\zeta(\vartheta)/\zeta_j(\vartheta) | \zeta_j(\vartheta) > 0\}$
            $y_j \leftarrow y_j + \psi(w_j - c^T \zeta_j)$
            break
        end if
    end for
    return $y_j$
end function
```