

1. Stato e Azioni

- **Stati (State):**

$$s = (r, t, p, v_{own}, v_{int}, pra)$$

Dove:

- r : Distanza tra il proprio velivolo (ownship) e l'intruso.
- t : Angolo relativo rispetto alla rotta del proprio velivolo.
- p : Angolo di direzione dell'intruso.
- v_{own} : Velocità del proprio velivolo.
- v_{int} : Velocità dell'intruso.
- pra : Risoluzione precedente (advisory).

- **Azioni (Actions):**

Set di azioni possibili: $\{COC, WL, WR, SL, SR\}$, che corrispondono a:

- **COC**: Clear of Conflict (nessuna manovra necessaria).
- **WL**: Virata leggera a sinistra.
- **WR**: Virata leggera a destra.
- **SL**: Virata forte a sinistra.
- **SR**: Virata forte a destra.

2. Funzione di Transizione

Definisce la probabilità di passare da uno stato s a uno stato successivo s' , dato un'azione a .
Calcola gli stati successivi possibili e le probabilità associate:

$$P(s'|s, a) = P(\text{ownship action}) \cdot P(\text{intruder action})$$

- Transizioni calcolate combinando:
 - Probabilità ownProbs e angoli ownTurns del velivolo proprio.
 - Probabilità intProbs e angoli intTurns dell'intruso.
-

3. Dinamiche dello Stato

Aggiorna lo stato in base alle azioni applicate e alle dinamiche:

$$x_{own} = x_{own} + v_{own} \cdot \Delta t$$

$$y_{own} = y_{own}$$

$$x_{int} = r \cdot \cos(t) + v_{int} \cdot \cos(p) \cdot \Delta t$$

$$y_{int} = r \cdot \sin(t) + v_{int} \cdot \sin(p) \cdot \Delta t$$

$$r' = \sqrt{(x_{int} - x_{own})^2 + (y_{int} - y_{own})^2}$$

$$t' = \text{normalize_angle}(\arctan 2(y_{int} - y_{own}, x_{int} - x_{own}) - \text{ownTurn})$$

$$p' = \text{normalize_angle}(p + \text{intTurn} - \text{ownTurn})$$

4. Funzione di Ricompensa

La funzione di ricompensa valuta la bontà di uno stato e un'azione:

$$R(s, a) = -\text{penalità}(r, \text{dCPA}, \text{tCPA}) - \text{penalità_alert}(ra)$$

Penalità sulla distanza:

- Quando $\tau = 0$:

$$\text{penalità} = \begin{cases} -1 & \text{se } r \leq 500 \\ -1 \cdot \exp\left(-\frac{r-500}{500}\right) & \text{altrimenti} \end{cases}$$

Penalità sul CPA (Closest Point of Approach):

- Penalità aggiuntiva quando il consiglio è COC:

$$\text{penalità}_{\text{CPA}} = \frac{1e - 2 \cdot \exp(-\text{dCPA}/500) \cdot \exp(-\text{tCPA}/10)}{\text{fattore}}$$

Penalità basate sull'advisory:

- Penalità se $ra \neq \text{COC}$:
 - Penalità per un alert forte:

$$\text{penalità}_{\text{alert}} = -2e - 3 \cdot \text{fattore}$$

- Penalità per inconsistenza con il consiglio precedente (*pra*):

$$\text{penalità}_{\text{conflitto}} = \begin{cases} -5e - 2 & \text{se } pra \text{ e } ra \text{ hanno senso opposto} \\ -1e - 3 & \text{se } ra \text{ rinforza } pra \\ -2e - 4 & \text{se } ra \text{ indebolisce } pra \end{cases}$$

Cosa fa compute_Qa()?

1. **Calcola il valore atteso di un'azione a per tutti gli stati s .**
2. Usa la rappresentazione matriciale delle probabilità di transizione per calcolare i valori futuri scontati, ottimizzando le operazioni.
3. Combina le **ricompense immediate** r e i **valori futuri scontati** $\gamma \cdot \text{trans} \cdot U$.

Formula Matematica:

Il calcolo segue questa formula:

$$Q_a(s) = R_a(s) + \gamma \sum_{s'} P(s'|s, a) \cdot U(s')$$

Qui:

- $R_a(s)$: Ricompensa immediata associata all'azione a nello stato s .
- γ : Fattore di sconto.
- $P(s'|s, a)$: Probabilità di transizione dallo stato s al prossimo stato s' dato che si esegue a .
- $U(s')$: Utilità (o valore) dello stato s' dalla precedente iterazione.

Spiegazione degli Argomenti:

1. **r (array) :**
 - È un vettore che contiene le ricompense immediate per ogni stato associato all'azione corrente.
 - Dimensione: $|S|$ (numero di stati).
 2. **gam (float) :**
 - È il fattore di sconto γ , che pesa quanto considerare importanti le ricompense future rispetto a quelle immediate.
 - Valore tipico: $0 < \gamma \leq 1$.
 3. **trans (csr_matrix) :**
 - Matrice sparsa che rappresenta la probabilità di transizione $P(s'|s, a)$ per una data azione.
 - Dimensione: $|S| \times |S|$, dove $|S|$ è il numero di stati.
 - Ogni riga corrisponde allo stato attuale s , e ogni colonna a uno stato successivo s' .
 4. **U (array) :**
 - È un vettore che contiene i valori degli stati dalla precedente iterazione dell'algoritmo (ad esempio, Value Iteration).
 - Dimensione: $|S|$.
-

Passaggi del Calcolo:

1. **Ricompense Immediate:**
 - Si parte con il vettore delle ricompense r , che rappresenta il valore immediato per ogni stato s associato all'azione considerata.
2. **Valore Futuro Scontato:**
 - La matrice di transizione `trans` viene moltiplicata per il vettore dei valori U usando il prodotto matrice-vettore `trans.dot(U)`. Questo calcola il valore atteso degli stati futuri pesato dalla probabilità di transizione.
 - Il risultato viene poi moltiplicato per γ per applicare il fattore di sconto.
3. **Somma dei Termini:**

- Si sommano le ricompense immediate r e i valori futuri scontati $\gamma \cdot \text{trans} \cdot U$.