

## Algorithms to Compute Appraisal Variables

---

**Algorithm 1** (Relevance)

---

```
1: function ISEVENTRELEVANT(Events  $\varepsilon_t$ )  
  
2:   Initialize graph  $\mathcal{G}_t$  with current mental state  $\mathcal{S}_t$ .  
  
3:    $\vec{g} \leftarrow \text{EXTRACTGOAL}(\mathcal{G}_t)$   
  
4:    $\mathcal{P}_t \leftarrow \text{EXTRACTPATHS}(\varepsilon_t, \vec{g})$   
  
5:   if ( $\mathcal{P}_t = \emptyset$ ) then  
6:     return 0  
7:   else  
8:      $\mathcal{U}_t \leftarrow \text{GETEVENTUTILITY}(\varepsilon_t, \vec{g})$   
9:     if ( $\mathcal{U}_t \geq \tau_e$ ) then  
10:      return ( $\mathcal{U}_t$ )  
11:    else  
12:      return 0  
13: end function
```

---

---

**Algorithm 2** (Desirability)

---

```
1: function ISEVENTDESIRABLE(MentalState  $\mathcal{S}_t$ , Event  $\varepsilon_t$ )

2:   Initialize graph  $\mathcal{G}_{t-1}$  with previous mental state  $\mathcal{S}_{t-1}$ .
3:   Initialize graph  $\mathcal{G}_t$  with current mental state  $\mathcal{S}_t$ .

4:    $\vec{g}_{t-1} \leftarrow \text{EXTRACTGOALS}(\mathcal{G}_{t-1})$ 
5:    $\mathcal{P}_{\vec{g}_{t-1}} \leftarrow \text{EXTRACTPATHS}(\mathcal{G}_{t-1}, \vec{g}_{t-1})$ 

6:    $\vec{g}_t \leftarrow \text{EXTRACTGOALS}(\mathcal{G}_t)$ 
7:    $\mathcal{P}_{\vec{g}_t} \leftarrow \text{EXTRACTPATHS}(\mathcal{G}_t, \vec{g}_t)$ 

8:   if ( $\varepsilon_t.type = \text{EXTERNAL}$ ) then
9:      $\chi \leftarrow \text{GETUTTERANCEUTILITY}(\mathcal{P}_{\vec{g}_t}, \mathcal{P}_{\vec{g}_{t-1}})$ 
10:     $\Lambda \leftarrow \text{GETACTIONUTILITY}(\mathcal{P}_{\vec{g}_t}, \mathcal{P}_{\vec{g}_{t-1}})$ 
11:     $\mathcal{E} \leftarrow \text{GETEMOTIONUTILITY}(\mathcal{P}_{\vec{g}_t}, \mathcal{P}_{\vec{g}_{t-1}})$ 

12:     $\vec{\omega} \leftarrow \text{GETUTILITYWEIGHTS}()$ 

13:     $\Delta U \leftarrow \frac{\omega_0 \cdot \chi + \omega_1 \cdot \Lambda + \omega_2 \cdot \mathcal{E}}{(\omega_0 + \omega_1 + \omega_2)}$ 

14:  else if ( $\varepsilon_t.type = \text{INTERNAL}$ ) then
15:     $\Delta U \leftarrow \text{GETMENTALSTATEUTILITY}(\mathcal{P}_{\vec{g}_t})$ 

16:  if ( $\Delta U \geq \tau$ ) then
17:    return TRUE
18:  else
19:    return FALSE
20: end function
```

---

---

**Algorithm 3** (Get Utterance Utility)

---

```
1: function GETUTTERANCEUTILITY(Paths  $\mathcal{P}_{\vec{g}_t}$ , Paths  $\mathcal{P}_{\vec{g}_{t-1}}$ )  
  
2:    $\mathcal{U}_{\vec{g}_t} \leftarrow \sum_{\mathcal{P}_{\vec{g}_t}^{U_i} \in \mathcal{P}_{\vec{g}_t}^U} \text{GETMENTALSTATEUTILITY}(\mathcal{P}_{\vec{g}_t}^{U_i})$   
3:    $\hat{\mathcal{U}}_{\vec{g}_t} \leftarrow \frac{\mathcal{U}_{\vec{g}_t}}{|\mathcal{P}_{\vec{g}_t}^U|}$   
4:    $\mathcal{U}_{\vec{g}_{t-1}} \leftarrow \sum_{\mathcal{P}_{\vec{g}_{t-1}}^{U_i} \in \mathcal{P}_{\vec{g}_{t-1}}^U} \text{GETMENTALSTATEUTILITY}(\mathcal{P}_{\vec{g}_{t-1}}^{U_i})$   
5:    $\hat{\mathcal{U}}_{\vec{g}_{t-1}} \leftarrow \frac{\mathcal{U}_{\vec{g}_{t-1}}}{|\mathcal{P}_{\vec{g}_{t-1}}^U|}$   
6:   return  $(\hat{\mathcal{U}}_{\vec{g}_t} - \hat{\mathcal{U}}_{\vec{g}_{t-1}})$   
7: end function
```

---

---

**Algorithm 4** (Get Action Utility)

---

```
1: function GETACTIONUTILITY(Paths  $\mathcal{P}_{\vec{g}_t}$ , Paths  $\mathcal{P}_{\vec{g}_{t-1}}$ )  
  
2:    $\theta \leftarrow \text{GETPRECONDITIONUTILITY}(\mathcal{P}_{\vec{g}_t}, \mathcal{P}_{\vec{g}_{t-1}})$   
3:    $\psi \leftarrow \text{GETPOSTCONDITIONUTILITY}(\mathcal{P}_{\vec{g}_t}, \mathcal{P}_{\vec{g}_{t-1}})$   
  
4:    $\vec{\omega} \leftarrow \text{GETUTILITYWEIGHTS}()$   
  
5:    $\Delta U \leftarrow \frac{\omega_0 \cdot \theta + \omega_1 \cdot \psi}{\omega_0 + \omega_1}$   
  
6:   return  $\Delta U$   
7: end function
```

---

---

**Algorithm 5** (Get Emotion Utility)

---

```

1: function GETEMOTIONUTILITY(Paths  $\mathcal{P}_{\tilde{g}_t}$ , Paths  $\mathcal{P}_{\tilde{g}_{t-1}}$ )

2:    $\mathcal{P}_{\tilde{g}_{t-1}}^{\mathcal{E}} \leftarrow \text{EXTRACTEMOTIONPATHS}(\mathcal{P}_{\tilde{g}_{t-1}})$ 
3:    $\mathcal{P}_{\tilde{g}_t}^{\mathcal{E}} \leftarrow \text{EXTRACTEMOTIONPATHS}(\mathcal{P}_{\tilde{g}_t})$ 

4:    $\mathcal{E}_{t-1}^{other} \leftarrow \text{EXTRACTOTHERSEMOTION}(\mathcal{P}_{\tilde{g}_{t-1}}^{\mathcal{E}})$ 
5:    $\mathcal{E}_t^{other} \leftarrow \text{EXTRACTOTHERSEMOTION}(\mathcal{P}_{\tilde{g}_t}^{\mathcal{E}})$ 

6:    $\Upsilon_{t-1} \leftarrow \text{GETSOMATICMARKERS}(\mathcal{E}_{t-1}^{other})$ 
7:    $\Upsilon_t \leftarrow \text{GETSOMATICMARKERS}(\mathcal{E}_t^{other})$ 

8:    $\Delta U \leftarrow \frac{\left( \left( \left( (\Upsilon_t^A - \Upsilon_{t-1}^A) \times 15 \right) + (\Upsilon_t^V - \Upsilon_{t-1}^V) \right) \times 15 \right) + (\Upsilon_t^S - \Upsilon_{t-1}^S) \right) \times 15 + (\Upsilon_t^I - \Upsilon_{t-1}^I)}{15^4}$ 

9:   return  $\Delta U$ 
10: end function

```

---

---

**Algorithm 6** (Get Precondition Utility)

---

```

1: function GETPRECONDITIONUTILITY(Paths  $\mathcal{P}_{\vec{g}_t}$ , Paths  $\mathcal{P}_{\vec{g}_{t-1}}$ )

2:    $ratio_{\vec{g}_t/\vec{g}_{t-1}}^{sat} \leftarrow ratio_{\vec{g}_t}^{sat/total} \leftarrow 0$ 

3:    $\mathcal{P}_{\vec{g}_t}^A \leftarrow \text{EXTRACTACTIONPATHS}(\mathcal{P}_{\vec{g}_t})$ 

4:   if ( $\vec{g}_t = \vec{g}_{t-1}$ ) then
5:     if ( $\mathcal{P}_{t-1}^A \neq \emptyset$ ) OR ( $\mathcal{P}_t^A \neq \emptyset$ ) then

6:        $\mathcal{P}_{\vec{g}_{t-1}}^A \leftarrow \text{EXTRACTACTIONPATHS}(\mathcal{P}_{\vec{g}_{t-1}})$ 

7:        $Precond_{\mathcal{P}_{\vec{g}_t}^A}^{sat} \leftarrow \text{CHECKPRECONDITIONS}(\mathcal{P}_{\vec{g}_t}^A)$ 
8:        $Precond_{\mathcal{P}_{\vec{g}_{t-1}}^A}^{sat} \leftarrow \text{CHECKPRECONDITIONS}(\mathcal{P}_{\vec{g}_{t-1}}^A)$ 

9:        $ratio_{\vec{g}_t/\vec{g}_{t-1}}^{sat} \leftarrow \omega_0 \cdot \frac{Precond_{\mathcal{P}_{\vec{g}_t}^A}^{sat}}{Precond_{\mathcal{P}_{\vec{g}_{t-1}}^A}^{sat} |_{\neq 0}} + \omega_1 \cdot \frac{Precond_{\mathcal{P}_{\vec{g}_t}^A}^{sat}}{Precond_{\mathcal{P}_{\vec{g}_{t-1}}^A}^{total} |_{\neq 0}}$ 

10:      return  $ratio_{\vec{g}_t/\vec{g}_{t-1}}^{sat}$ 
11:    else
12:      return 0
13:  else
14:    if ( $\mathcal{P}_t^A \neq \emptyset$ ) then
15:       $Precond_{\mathcal{P}_{\vec{g}_t}^A} \leftarrow \text{CHECKPRECONDITIONS}(\mathcal{P}_{\vec{g}_t}^A)$ 

16:       $ratio_{\vec{g}_t}^{sat/total} \leftarrow \frac{Precond_{\mathcal{P}_{\vec{g}_t}^A}^{sat}}{Precond_{\mathcal{P}_{\vec{g}_t}^A}^{total} |_{\neq 0}}$ 

17:      return  $ratio_{\vec{g}_t}^{sat/total}$ 
18:    else
19:      return 0
20: end function

```

---

---

**Algorithm 7** (Check Preconditions)

---

```
1: function CHECKPRECONDITIONS(Paths  $\mathcal{P}_{\vec{g}}^A$ )
2:    $count_{precond}^{sat} \leftarrow count_{precond}^{total} \leftarrow 0$ 
3:    $\Theta_{\vec{g}} \leftarrow \text{EXTRACTPRECONDITIONS}(\mathcal{P}_{\vec{g}}^A)$ 
4:   for each  $\theta_{\vec{g}}^i \in \Theta_{\vec{g}}$  do
5:     if (ISSATISFIED( $\theta_{\vec{g}}^i$ )) then
6:        $count_{precond}^{sat} \leftarrow count_{precond}^{sat} + 1$ 
7:        $count_{precond}^{total} \leftarrow count_{precond}^{total} + 1$ 
8:   return  $\langle count_{precond}^{sat}, count_{precond}^{total} \rangle$ 
9: end function
```

---

---

**Algorithm 8** (Get Postcondition Utility)

---

```
1: function GETPOSTCONDITIONUTILITY(Paths  $\mathcal{P}_{\vec{g}_t}$ , Paths  $\mathcal{P}_{\vec{g}_{t-1}}$ )
2:    $\mathcal{P}_{\vec{g}_t}^A \leftarrow \text{EXTRACTACTIONPATHS}(\mathcal{P}_{\vec{g}_t})$ 
3:   if ( $\vec{g}_t = \vec{g}_{t-1}$ ) then
4:     if ( $\mathcal{P}_{t-1}^A \neq \emptyset$ ) OR ( $\mathcal{P}_t^A \neq \emptyset$ ) then
5:        $\mathcal{P}_{\vec{g}_{t-1}}^A \leftarrow \text{EXTRACTACTIONPATHS}(\mathcal{P}_{\vec{g}_{t-1}})$ 
6:        $\Psi_{\vec{g}_t} \leftarrow \text{EXTRACTPOSTCONDITIONS}(\mathcal{P}_{\vec{g}_t}^A)$ 
7:        $\Psi_{\vec{g}_{t-1}} \leftarrow \text{EXTRACTPOSTCONDITIONS}(\mathcal{P}_{\vec{g}_{t-1}}^A)$ 
8:       if (ISSATISFIED( $\Psi_{\vec{g}_t}$ )) then
9:         if (ISSATISFIED( $\Psi_{\vec{g}_{t-1}}$ )) then
10:          return  $\gamma^t$  ▷ Discount rate:  $0 \leq \gamma \leq 1$ 
11:         else
12:          return 1
13:       else
14:         if ISSATISFIED( $\Psi_{\vec{g}_{t-1}}$ ) then
15:          return -1
16:         else
17:          return 0
18: end function
```

---

---

**Algorithm 9** (Expectedness)

---

```
1: function ISEVENTEXPECTED(MentalStates  $\mathcal{S}_t$ , Event  $\varepsilon_t$ )

2:   Initialize graph  $\mathcal{G}_{t-1}$  with previous mental state  $\mathcal{S}_{t-1}$ .
3:   Initialize graph  $\mathcal{G}_t$  with current mental state  $\mathcal{S}_t$ .

4:    $\vec{g}_{t-1} \leftarrow \text{EXTRACTGOALS}(\mathcal{G}_{t-1})$ 
5:    $\vec{g}_t \leftarrow \text{EXTRACTGOALS}(\mathcal{G}_t)$ 

6:    $\mathcal{P}_{t-1} \leftarrow \text{EXTRACTPATHS}(\mathcal{G}_{t-1}, \vec{g}_{t-1})$ 
7:    $\mathcal{P}_t \leftarrow \text{EXTRACTPATHS}(\mathcal{G}_t, \vec{g}_t)$ 

8:   if ( $\vec{g}_t \neq \vec{g}_{t-1}$ ) then
9:     if (ISACHIEVED( $g_{t-1}^c$ )) then
10:      if ( $\neg \text{ISACHIEVED}(g_{t-1}^p)$ ) then
11:         $\mathcal{P}'_{t-1} \leftarrow \text{EXTRACTPATHS}(\mathcal{G}_{t-1}, \vec{g}_t)$ 
12:        if ( $\mathcal{P}'_{t-1} = \emptyset$ ) then
13:          return FALSE
14:        else
15:           $\mathcal{U}'_{t-1} \leftarrow \text{GETUTILITIES}(\mathcal{G}_{t-1}, \vec{g}_t)$ 
16:           $\mathcal{U}_{t-1} \leftarrow \text{GETUTILITIES}(\mathcal{G}_{t-1}, \vec{g}_{t-1})$ 
17:          if ( $\mathcal{U}'_{t-1} - \mathcal{U}_{t-1} \geq \tau_e$ ) then
18:            return TRUE
19:          else
20:            return FALSE
21:        else  $\triangleright$  Collaborative and personal goals have achieved.
22:        return TRUE
23:      else  $\triangleright$  Collaborative goal has not achieved.
24:      return FALSE
25:    else  $\triangleright$  Goals have not changed.
26:    return TRUE
27: end function
```

---

---

**Algorithm 10** (Likelihood)

---

```

1: function GETEVENTSLIKELIHOOD(MentalStates  $\mathcal{S}_t$ , Event  $\varepsilon_t$ )

2:    $\mathcal{G}_t \leftarrow \text{UPDATEGRAPH}(\mathcal{S}_t, \varepsilon_t)$ 
3:    $\vec{g}_t \leftarrow \text{EXTRACTGOALS}(\mathcal{G}_t)$ 
4:    $\mathcal{P}_t \leftarrow \text{EXTRACTPATHS}(\mathcal{G}_t, \vec{g}_t)$ 

5:   if ( $\mathcal{P}_t = \emptyset$ ) then
6:     return 0
7:   else
8:     for each  $\mathcal{P}_t^i \in \mathcal{P}_t$  do
9:       for each  $\varepsilon_{t+1}^j \in \varepsilon_{t+1}$  do
10:        
$$\vec{P}(\varepsilon_{t+1}^j | \mathcal{F}(\mathcal{P}_t^i)) = \frac{\vec{P}(\mathcal{F}(\mathcal{P}_t^i) | \varepsilon_{t+1}^j) \vec{P}(\varepsilon_{t+1}^j)}{\sum_{e_{t+1}^k \in \varepsilon_{t+1}^j} \vec{P}(\mathcal{F}(\mathcal{P}_t^i) | e_{t+1}^k)}$$

11:       return  $\vec{P}(\varepsilon_{t+1} | \mathcal{F}(\mathcal{P}_t^i))$ 
12: end function

```

---



---

**Algorithm 11** (Controllability)

---

```

1: function ISEVENTCONTROLLABLE(MentalStates  $\mathcal{S}_t$ , Event  $\varepsilon_t$ )

2:    $\alpha_{self/other}^{agency} \leftarrow \beta_{self/other}^{autonomy} \leftarrow 0$ 

3:    $\kappa_{satisfied/total}^{preconditions} \leftarrow \lambda_{succeeded/failed}^{predecessors} \leftarrow \mu_{available/required}^{inputs} \leftarrow 0$ 

4:   Initialize graph  $\mathcal{G}_t$  with current mental state  $\mathcal{S}_t$ .

5:    $\vec{g}_t \leftarrow \text{EXTRACTGOALS}(\mathcal{G}_t)$ 
6:    $\mathcal{P}_{\vec{g}_t} \leftarrow \text{EXTRACTPATHS}(\mathcal{G}_t, \vec{g}_t)$ 

7:    $\mathcal{P}_{\vec{g}_t}^A \leftarrow \text{EXTRACTACTIONPATHS}(\mathcal{P}_{\vec{g}_t})$ 
8:    $\varphi_{\vec{g}_t}^A \leftarrow \text{GETRECIPEAPPLICABILITY}(\mathcal{P}_{\vec{g}_t}^A)$ 

9:    $\mathcal{P}_{\vec{g}_t}^{\mathcal{E}} \leftarrow \text{EXTRACTEMOTIONPATHS}(\mathcal{P}_{\vec{g}_t})$ 

10:   $\mathcal{E}_t^{other} \leftarrow \text{EXTRACTOTHERSEMOTION}(\mathcal{P}_{\vec{g}_t}^{\mathcal{E}})$ 

11:   $\mathcal{I}_{other}^{emotion} \leftarrow \text{GETEMOTIONINTENSITY}(\mathcal{E}_t^{other})$ 

12:   $\Upsilon_t \leftarrow \text{GETSOMATICMARKERS}(\mathcal{E}_t^{other})$ 
13:   $\mathcal{V}_t^{\mathcal{E}} \leftarrow \text{GETEMOTIONVALUE}(\Upsilon_t, \varepsilon_t)$ 

14:   $Agency_{\vec{g}_t} \leftarrow \text{GETAGENCYVALUE}(\mathcal{P}_{\vec{g}_t})$ 
15:   $\alpha_{self/other}^{agency} \leftarrow Agency_{\vec{g}_t}^{self} / Agency_{\vec{g}_t}^{other}$ 

16:   $\beta_{self/other}^{autonomy} \leftarrow \text{GETAUTONOMYVALUE}(\mathcal{P}_{\vec{g}_t}^{self}) / \text{GETAUTONOMYVALUE}(\mathcal{P}_{\vec{g}_t}^{other})$ 

17:   $Precond_{\mathcal{P}_{\vec{g}_t}^A} \leftarrow \text{CHECKPRECONDITIONS}(\mathcal{P}_{\vec{g}_t}^A)$ 
18:   $\kappa_{satisfied/total}^{preconditions} \leftarrow \frac{Precond_{\mathcal{P}_{\vec{g}_t}^A}^{satisfied}}{Precond_{\mathcal{P}_{\vec{g}_t}^A}^{total}}$ 

19:   $Predecessors_{\mathcal{P}_{\vec{g}_t}^A} \leftarrow \text{CHECKPREDECESSORS}(\mathcal{P}_{\vec{g}_t}^A)$ 
20:   $\lambda_{succeeded/total}^{predecessors} \leftarrow \frac{Predecessors_{\mathcal{P}_{\vec{g}_t}^A}^{succeeded}}{Predecessors_{\mathcal{P}_{\vec{g}_t}^A}^{total}}$ 

21:   $Inputs_{\mathcal{P}_{\vec{g}_t}^A} \leftarrow \text{CHECKINPUTS}(\mathcal{P}_{\vec{g}_t}^A)$ 
22:   $\mu_{available/required}^{inputs} \leftarrow \frac{Inputs_{\mathcal{P}_{\vec{g}_t}^A}^{available}}{Inputs_{\mathcal{P}_{\vec{g}_t}^A}^{required}}$ 

```

---

---

```

23:  $\mathcal{U}_t \leftarrow \frac{\omega_0 \cdot \alpha_{self/other}^{agency} + \omega_1 \cdot \beta_{self/other}^{autonomy}}{\omega_0 + \omega_1}$ 
24:  $+ \frac{\omega_2 \cdot \kappa_{satisfied/total}^{preconditions} + \omega_3 \cdot \lambda_{succeeded/total}^{predecessors} + \omega_4 \cdot \mu_{available/required}^{inputs}}{\omega_2 + \omega_3 + \omega_4}$ 
25:  $+ \frac{\omega_5 \cdot \mathcal{I}_{other}^{emotion} + \omega_6 \cdot \varphi_{\vec{g}_t}^A + \omega_7 \cdot \mathcal{V}_t^E}{\omega_5 + \omega_6 + \omega_7}$ 
26: if ( $\mathcal{U}_t \geq \tau_e$ ) then
27:     return TRUE
28: else
29:     return FALSE
30: end function

```

---



---

**Algorithm 12** (Check Predecessors)

---

```

1: function CHECKPREDECESSORS(Paths  $\mathcal{P}_{\vec{g}}^A$ )
2:    $count_{predecessor}^{succeeded} \leftarrow count_{predecessor}^{total} \leftarrow 0$ 
3:    $\Phi_{\vec{g}} \leftarrow \text{EXTRACTPREDECESSORS}(\mathcal{P}_{\vec{g}}^A)$ 
4:   for each  $\phi_{\vec{g}}^i \in \Phi_{\vec{g}}$  do
5:     if (ISUCCEEDED( $\phi_{\vec{g}}^i$ )) then
6:        $count_{predecessor}^{succeeded} \leftarrow count_{predecessor}^{succeeded} + 1$ 
7:        $count_{predecessor}^{total} \leftarrow count_{predecessor}^{total} + 1$ 
8:   return  $\langle count_{predecessor}^{succeeded}, count_{predecessor}^{total} \rangle$ 
9: end function

```

---



---

**Algorithm 13** (Check Inputs)

---

```

1: function CHECKINPUTS(Paths  $\mathcal{P}_{\vec{g}}^A$ )
2:    $count_{input}^{available} \leftarrow count_{input}^{required} \leftarrow 0$ 
3:    $\mathcal{X}_{\vec{g}} \leftarrow \text{EXTRACTINPUTS}(\mathcal{P}_{\vec{g}}^A)$ 
4:   for each  $\chi_{\vec{g}}^i \in \mathcal{X}_{\vec{g}}$  do
5:     if (ISUCCEEDED( $\chi_{\vec{g}}^i$ )) then
6:        $count_{input}^{available} \leftarrow count_{input}^{available} + 1$ 
7:        $count_{input}^{required} \leftarrow count_{input}^{required} + 1$ 
8:   return  $\langle count_{input}^{available}, count_{input}^{required} \rangle$ 
9: end function

```

---

---

**Algorithm 14** (Get Agency Value)

---

```
1: function GETAGENCYVALUE(Paths  $\mathcal{P}_{\bar{g}}^A$ )
2:    $count_{responsibility}^{self} \leftarrow count_{responsibility}^{other} \leftarrow 0$ 
3:    $\Theta_{\bar{g}} \leftarrow \text{EXTRACTPRECONDITIONS}(\mathcal{P}_{\bar{g}}^A)$ 
4:   for each  $\theta_{\bar{g}}^i \in \Theta_{\bar{g}}$  do
5:     if ( $\text{GETRESPONSIBLE}(\theta_{\bar{g}}^i) = \text{SELF}$ ) then
6:        $count_{responsibility}^{self} \leftarrow count_{responsibility}^{self} + 1$ 
7:     else
8:        $count_{responsibility}^{other} \leftarrow count_{responsibility}^{other} + 1$ 
9:   return  $\langle count_{responsibility}^{self}, count_{responsibility}^{other} \rangle$ 
10: end function
```

---

---

**Algorithm 15** (Get Autonomy Value)

---

```
1: function GETAUTONOMYVALUE(Paths  $\mathcal{P}_{\bar{g}}^A$ )
2:    $\mathcal{A} \leftarrow \text{EXTRACTACTION}(\mathcal{P}_{\bar{g}}^A)$ 
3:    $\mathcal{R}_{\mathcal{A}} \leftarrow \text{GETRESPONSIBLE}(\mathcal{A})$ 
4:    $\mathcal{M}_{\mathcal{R}_{\mathcal{A}}} \leftarrow \text{GETMOTIVE}(\mathcal{R}_{\mathcal{A}})$ 
5:   if ( $\mathcal{M}_{\mathcal{R}_{\mathcal{A}}} \neq \emptyset$ ) then
6:     return MAX
7:   else
8:     return MIN
9: end function
```

---