

# Pseudocode for Project

jmuguirra

June 2022

## 1 Introduction

---

**Algorithm 1** OUR\_ALGORITHM( $((Q, E), W, V, q_0, Q_{goal}, v_0)$ )

---

```
while  $q_{current} \neq Q_{goal}$  do
   $p_{current} \leftarrow \text{VAMP\_BACKCHAIN}((Q, E), W, V, q_0, Q_{goal}, v_0)$ 
  while obstruction = False do
     $q_{current} \leftarrow \text{STEP}(q_{current}, p_{current})$ 
    UPDATE_VISION
    obstruction = FIND_OBSTRUCTION( $q_{current}, p_{current}$ )
```

---

---

**Algorithm 2** VAMP\_BACKCHAIN( $((Q, E), W, V, q_0, Q_{goal}, v_0)$ )

---

```
1: procedure VAVP( $q, R, v, O = \emptyset$ )
2:    $p_{vis} \leftarrow \text{TOURIST}(q, R, v)$ 
3:   if  $p_{vis} \neq \text{Failed}$  then return  $p_{vis}$ 
4:    $O_{new} = O \cup R$ 
5:    $p_{relaxed} \leftarrow \text{TOURIST}(q, R, v, relaxed = \text{true}, O = O_{new})$ 
6:   if  $p_{relaxed} \neq \text{Failed}$  then
7:      $p_{vis} \leftarrow \text{VAVP}(q, S(p_{relaxed}) \setminus v, v, O = O_{new})$ 
8:     if  $p_{vis} \neq \text{Failed}$  then return  $p_{vis}$ 
9:   return Failed
10:
11:  $p \leftarrow []$ ;  $v \leftarrow v_0$ ;  $q \leftarrow q_0$ 
12: while True do
13:    $p_{final} \leftarrow \text{VAMP\_PATH\_VIS}(q, Q_{goal}, v)$ 
14:   if  $p_{final} \neq \text{Failed}$  then return  $p + p_{final}$ 
15:    $p_{relaxed} \leftarrow \text{VAMP\_PATH\_VIS}(q, Q_{goal}, v, relaxed = \text{True})$ 
16:    $p_{vis} \leftarrow \text{VAVP}(q, S(p_{relaxed}) \setminus v, v)$ 
17:   if  $p_{vis} = \text{Failed}$  then  $p_{vis} \leftarrow \text{TOURIST}(q, W \setminus v, v)$ 
18:   if  $p_{vis} = \text{Failed}$  then return Failed
19:    $p \leftarrow p + p_{vis}$ ;  $v \leftarrow \cup V(p_{vis})$ ;  $q \leftarrow p_{vis}[-1]$ 
```

---

---

**Algorithm 3** VAMP\_STEP\_VIS( $((Q, E), V, q_0, Q_{goal}, v_0, H = 0, relaxed = False, O = \emptyset)$ )

---

$s_0 \leftarrow q_0$   
**if** relaxed **then**  
     $A(q) \leftarrow \{q' | (q, q') \in E \text{ and } S(q, q') \cap O = \emptyset\}$   
     $C(q, q') \leftarrow \|q - q'\|_2$  (**if**  $S(q, q') \subseteq (v_0 \cup V(q))$  **else**  $|S(q, q') \setminus (v_0 \cup V(q))|$ )  
**else**  
     $A(q) \leftarrow \{q' | (q, q') \in E \text{ and } S(q, q') \subseteq (v_0 \cup V(q))\}$   
     $C(q, q') \leftarrow \|q - q'\|_2$   
 $T(q, q') \leftarrow q'$   
 $p_{temp} \leftarrow A^*(s_0, Q_{goal}, A, T, H, C)$   
**if**  $p_{temp} = Failed$  **then**  
     $p_{temp} \leftarrow \text{VAMP\_PATH\_VIS}(q, Q_{goal}, v, relaxed, O \setminus OBS)$   
    **if**  $p_{temp} = Failed$  **then return** Failed  
**else**  
     $q_{attach} \leftarrow \text{SAMPLE\_ATTACHMENT}(p_{temp})$   
     $P_{temp} \leftarrow \text{VAMP\_BACKCHAIN}(q_0, q_{attach})$   
     $q_{place} \leftarrow \text{SAMPLE\_DETACHMENT}()$   
     $P_{temp} \leftarrow p_{temp} + \text{VAMP\_BACKCHAIN}(q_{attach}, q_{place})$   
**return**  $p_{temp}$

---



---

**Algorithm 4** TOURIST( $((Q, E), V, q_0, R, v_0, relaxed = False, O = \emptyset)$ )

---

$H(q) = \min_{x \in V(q)} F(x)$   
**return** VAMP\_PATH\_VIS( $((Q, E), V, q_0, \lambda q. (V(q) \cap R) \neq \emptyset, v_0, H, relaxed, O)$ )

---