

---

**ALGORITHM 1:** Get inter-arrival distribution
 

---

```

input : Log Traces  $T$ 
input : BPMN Model  $M$ 
 $Sa \leftarrow \text{get\_starting\_activities}(M)$ ;
 $St \leftarrow \emptyset$ ; // first activities start times list
 $\Delta St \leftarrow \emptyset$ ; // time deltas list
foreach  $t \in T$  do
  foreach  $e \in t$  do
    if  $\exists \lambda(e) \in Sa$  then
       $St \leftarrow St \cup \{v(e)\}$ ;
    end
  end
end
 $St \leftarrow \text{sorted}(St)$ ; // sorting start times
for  $i = 0$  to  $(\text{length}(St) - 1)$  do
   $\Delta St \leftarrow \Delta St \cup \{St_{i+1} - St_i\}$ ;
end
return  $\text{get\_probability\_distribution}(\Delta St)$ 

```

---



---

**ALGORITHM 2:** Get activities duration distribution
 

---

```

input : Log Traces  $T$ 
input : BPMN Model  $M$ 
 $Sa \leftarrow \text{get\_activities}(M)$ ;
 $DDa \leftarrow \emptyset$ ; // Set of activities probability distributions
foreach  $a \in Sa$  do
   $\Delta a \leftarrow \text{get\_durations}(M, a)$ ;
   $DDa \leftarrow DDa \cup \text{get\_probability\_distribution}(\Delta a)$ 
end
return  $DDa$ 

```

---

---

**ALGORITHM 3:** Paths probabilities definition
 

---

```

input  : BPMN Model  $M$ 
 $G \leftarrow \text{get\_process\_gates}(M)$  ;           // set of XOR gates
foreach  $g \in G$  do
     $Pt \leftarrow \text{get\_gate\_paths}(M, g)$  ;           // set of gate paths
     $To \leftarrow 0$  ;           // count of total frequency of paths execution
    foreach  $p \in Pt$  do
         $To \leftarrow To + \text{get\_ocurrences}(M, p)$ ;
    end
    foreach  $p \in Pt$  do
         $prob \leftarrow (\text{get\_ocurrences}(M, p) / To)$ ;
         $\text{set\_path\_probability}(M, p, prob)$  ;    // modify the path probability
    end
end
  
```

---