Algorithm. 1

INITIATE PHASE##

Initiate forwardPeriod;
Initiate the set of cars C= {c1, c2, …};
Initiate the set of state= {Accident, receiveMsgOnStop, receiveMsgOnRoute};
For each c_i in C do
    Initiate Path p_i = {s_1, s_2, …, destination};// s_j represents a unique street in the city map
End for

RUNTIME PHASE##

```
1: While (true) do
2:  Switch (state):
3:  Accident:
4:        sendMessage (sj, accidentTime, accidentDuration, accidentType, accidentCoords);
5:  receiveMsgOnStop:// this set of cars includes the cars which stopped due to the accident
6:    if (possible_to_change_path)
7:          set path=shortest_path_to_destination;
8:          change route;
9:    end if;
10:   while (currentTime+ forwardPeriod<accidentTime+ accidentDuration) do
11:         pause(forwardPeriod+backoff);
12:         sendMessage(sj), accidentTime, accidentDuration, accidentType, accidentCoords);
13:   end while;
14:  receiveMsgOnRoute:
15:   if (Msg.sj in my Remaining_Path)
16:    if (Msg.accidentType==Easy)
17:     Calculate Distance_to_sj
18:     Switch (Distance_to_sj)
19:       Large:
20:        Ignore the message;
21:       Medium:
22:        Ignore the message;
23:       Small:
24:        set path=shortest_path_to_destination;
25:        change route;
26:      while (currentTime+ forwardPeriod<accidentTime+ accidentDuration) do
27:          sendMessage (sj, accidentTime, accidentDuration, accidentType, accidentCoords);
             pause(forwardPeriod+backoff);
28:      end while;
29:    else
30:       set path=shortest_path_to_destination;
31:       change route;
32:     while (currentTime+ forwardPeriod<accidentTime+ accidentDuration) do
33:        sendMessage (sj, accidentTime, accidentDuration, accidentType, accidentCoords);
34:        pause(forwardPeriod+backoff);
35:     end while;
36:    end if;
37:   else
38:    if (Msg.accidentType==Hard)
39:     Calculate Distance_to_s
40:     Switch (Distance_to_sj)
41:       Large:
42:        sendMessage (sj, accidentTime, accidentDuration, accidentType, accidentCoords);
43:       Medium:
44:        sendMessage (sj, accidentTime, accidentDuration, accidentType, accidentCoords);
45:       Small:
46:         Ignore the message;
47:    else
48:        Ignore the message;
49:    end if;
50:   end if;
51:  default:
52:    keep tracking;
53:  end Switch;
54: End while;
```