



Transport and Telecommunication, 2020, volume 21, no. 2, 110–118  
Transport and Telecommunication Institute, Lomonosova 1, Riga, LV-1019, Latvia  
DOI 10.2478/ttj-2020-0008

## AUCTION BASED ALGORITHM FOR A SMART JUNCTION WITH SOCIAL PRIORITIES

Orly Barzilay<sup>1</sup>, Amit Giloni<sup>2</sup>, Nadav Voloch<sup>2</sup> and Orna Lavi Steiner<sup>3</sup>

<sup>1</sup>The Academic College of Tel-Aviv – Yaffo, Rabenu Yeruham St. 2, Yaffo. 64162, Israel  
orlyba@mta.ac.il

<sup>2</sup>Ben-Gurion University of the Negev, P.O.B. 653, Beer-Sheva, 8410501, Israel  
hacmona@post.bgu.ac.il  
voloch@post.bgu.ac.il

<sup>3</sup>Ruppin Academic Center, Emek Hefer, Israel  
orna.steiner@gmail.com

Smart devices and their connections to the Internet of Things (IoT) have been the subject of many papers in the past decade. In the context of IoT in transportation, one feature is the smart junction. This research deals with this junction, where several cars approach the intersection from different directions, and a smart traffic light must decide regarding the time intervals of red and green light in each direction. Our novel approach is based not only on the number of vehicles in each lane, but also on the social characteristics of the passengers (e.g. a handicapped person, a driver with no previous traffic violations). These factors will be gleaned from IoT network sources on cars, traffic lights, individuals, municipality data, and more. In this paper, we suggest using a VCG (Vickrey-Clarke-Groves) auction mechanism for the intersection scheduling, combining the social characteristic with a benefit parameter that expresses the passenger's subjective perception of the importance of crossing the intersection as soon as possible. Our simulation results show the efficiency of the suggested protocol and demonstrate how the intersection scheduling depends on the passengers' preferences, as well as on their social priorities.

**Keywords:** Smart junction, Internet of Things (IoT), Real-Time algorithms, Social dilemmas in IoT, VCG auction

### 1. Introduction

In the past decade, smart junction algorithms have been developed to achieve better traffic congestion management, reduced driver's frustration and better time management (Khalid *et al.*, 2008).

Several algorithms propose traffic navigation within a junction based on conflict sides' negotiation, while considering extreme cases and scenarios, such as emergency vehicles (Collotta *et al.*, 2013; Pau and Scata, 2014; Younes and Boukerche, 2017). "Conflict side" is defined in the next section. The authors in (Veldscholten, 2015) suggest that better junction traffic load management can reduce accidents by lowering the driver's stress caused by excessive waiting times at the junction. The idea that driver characteristics that are beyond the driving cognitive abilities affect driving safety is supported by the work in (Reynolds and Ceranic, 2007).

In (Barzilay *et al.*, 2018a), the authors present a new algorithm that controls traffic light timing based on the number of vehicles per each conflict side, integrated with social priorities. A survey showed that people have social preferences for different vehicles types. The highest social preference is allocated to emergencies, such as rescue vehicles, and a moderate preference is allocated to public transport in comparison to private vehicles.

The smart junction with social priorities algorithm was elaborated upon in (Barzilay *et al.*, 2018b). The new algorithm presented there includes more realistic characteristics, taking into consideration different vehicle and junction parameters with non-obvious social preferences, such as prioritizing disabled passengers or passengers with no traffic violations. Green light duration is determined by the traffic congestion, while the conflict side's scheduling is determined by the social priority preferences summed per each conflict side.

Simulation results showed that a smart junction with social priorities evacuated the junction vehicles faster than a regular junction. In addition, it is suggested that green light scheduling prioritized by social parameters can serve as a tool to encourage safer road behaviour.

An important issue that was not considered in the works on using social priority for green light scheduling, is how important is it for a passenger to cross the junction rapidly. For example, a passenger late for an important work meeting will find intersection delay far more troubling than another on a shopping trip. It could be argued that using a social parameter to promote road safety is influenced by the passenger's benefit from prioritizing his vehicle within the junction.

Since passenger benefit is highly subjective, we seek authentic passengers' reports in this regard. In this study, we suggest using the VCG auction protocol (Groves, 1973), which ensures truthful reporting of the passengers' preferences for reaching optimal traffic scheduling.

An auction-based tool for managing autonomous vehicles waiting at intersections is suggested in (Carlinom *et al.*, 2013). A “wallet” system is used for automatic bidding based on trip characteristics, a driver-specific budget and the remaining distance to the destination. Using several representative city networks, the authors concluded that the contribution of the system bids for saving time is unclear.

The bids in our model are specified using special "transportation points" that are given to a vehicle for a given time period (e.g., per month), where vehicles with higher social priority obtain a larger number of points in each time period, and thus, will be able to use more points while travelling.

**2. Method**

**2.1. Definition of social priority and vehicle benefit**

The current study extends an algorithm for managing smart junctions with social priorities suggested in previous research (Barzilay *et al.*, 2018b; Carlinom *et al.*, 2013). A new "benefit" parameter is added to the algorithm, representing the importance of promoting the location ranking of the vehicle at the intersection. In other words, a vehicle's benefit parameter expresses the passenger's subjective perception of the importance of crossing the intersection as soon as possible.

Based on the methodology developed in previous research (Barzilay *et al.*, 2018b; Carlinom *et al.*, 2013), each vehicle is attributed a social priority value randomly selected within the range of 1 to 10, where 10 is the highest priority value. Each vehicle receives a certain amount of points that can be invested to advance the location ranking of the vehicle at the intersection. The points are calculated by multiplying the value of the social priority by one hundred, resulting in a range of 100-900 points.

The vehicle benefit represents the utility of promoting the vehicle location at the intersection. The benefit value is calculated by randomly selecting a value within the range of 0 to 0.2 and multiplying this value by the vehicle total points. In order to prevent the situation where all the points are expended at the first intersection, a range of 0-0.2 (a mean of 0.1) was designed to create a distributed point investment scheme between 10 intersections. The simulation in this study was done on a single intersection, and we gave the option for a driver to divide his or her social points, based on his benefit, between 10 junctions that simulate a real-life route. The idea behind the use of a vehicle social preference and personal benefit is that, at the level of the individual vehicle, the higher the social preference and the personal benefit the sooner the car will pass the junction.

**2.2. Java based Simulator for a smart junction**

A prototype Java-based integrated simulator, developed in previous research (Carlinom *et al.*, 2013), was enhanced for this study. Figure 1 presents the junction structure. The junction contains 12 lanes grouped into four conflict sides. A conflict side is all the lanes in all the directions of a traffic junction that can move simultaneously without obstructing one another.

The four-conflict-side scheme presented in Figure 1 is: Conflict side 1: lanes C+G, Conflict side 2: lanes A+E, Conflict side 3: lanes H+D, Conflict side 4: lanes F+B.

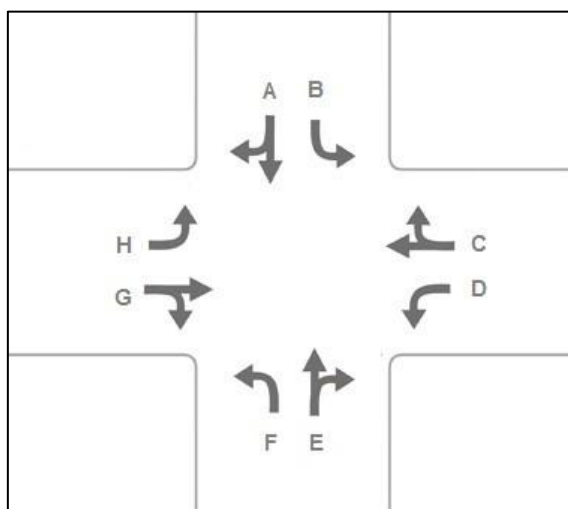


Figure 1. The lanes of the junction

The simulation is based on 1600 cars equally scattered between the different lanes. The vehicles' physical dimensions are fixed while the values of the social priority and benefit values are randomly assigned.

Each vehicle has a social priority value randomly selected within the range of 1 to 10, where 10 is the highest priority value. Each vehicle has a benefit value based on a random value generated from a range of 0 to 0.2.

The social priority value represents the priority given to a vehicle based on the social properties of the driver such as a driver without traffic violations during the past 5 years or a driver that has cancer. In a given junction conflict side, the social priority value is the summation of all vehicles' social priority values.

In this simulation, we eliminated emergency vehicles used in previous research in order to emphasize the impact of the benefit factor.

In order to emphasize the benefit impact, we set a fixed value for several junction and vehicle physical parameters. The junction physical properties are set to 15 meters for the straight direction and 10 meters for the corner directions. The maximal green light time is 60 seconds. For all the vehicles in the simulation junction the length of the vehicle is 4 meters, the speed is 2.7 meters per second and there is one person in the car. For this simulation, we used 1600 vehicles equally spread across the four conflict sides. Each vehicle can promote its lane location by using its benefit value. The integration of a single benefit values in a given conflict side is implemented using the VCG auction protocol, detailed in the next section.

The simulation ran until all vehicles passed the intersection. The results described in part 3 are based on a several runs of the simulation – runs that gave similar results.

### 2.3. The VCG Auction protocol

The auction type used in this study is the VCG protocol, which is used to determine which conflict side will get the green light and what price each vehicle will pay.

In each simulation, each vehicle gives a bid (benefit), which represents the price the vehicle agrees to pay in order to promote its location.

Then, the VCG protocol is used to determine the payments of each car in the winning side: each car that influences the winning of a conflict side (i.e., without this car, another conflict side would be the winner), pays the difference between total bids of its conflict side and the total bids of the second conflict side.

The VCG protocol is truthful: each car is motivated to bid its real value for proceeding in the junction. In addition, this protocol is "Individual Rational": each car does not pay more than its bid.

#### *VCG algorithm and starvation*

A starved conflict side is a conflict side which does not receive a green light for a long time. In order to avoid starvation, a maximal number of 5 green lights since the last time a conflict side received a green light is defined. A starved conflict side receives a green light regardless of other calculations.

### 2.4. The algorithm description and its complexity analysis

Per each conflict side 3 attributes are defined:

- CSS (Conflict Side Starvation value): the number of green lights passed since the last time the conflict side received a green light.
- CSB (Conflict Side Benefit value): the sum of benefit values of all vehicles in the conflict side.
- CST (Conflict Side Time): the amount of green light time set for the conflict side.

#### *Allocating a green light to a conflict side algorithm*

Per each time stamp (described in "VCG algorithm and starvation" section), a starved conflict side (if such exists) receives the green light. If no starved conflict sides exist, the green light is allocated to the conflict sides with the maximal CSB value.

For the algorithm  $n$  is the total number of vehicles in the junction. Steps 1-3 are initializations of the variables of the maximum benefit, the chosen conflict side and the maximum starvation. The complexity of these steps is  $O(1)$ . Steps 4 and 6 go iteratively over all the conflict sides, which we know are four. Therefore, the complexity of these steps is  $O(1)$  as well. Step 5 returns if a conflict side exists. Step 7 returns the chosen conflict side. Both of these steps' complexity is  $O(1)$ . Hence, the total complexity of the algorithm is  $O(1)$ .

The algorithm is as follows:

**OpenGreenLight** (*Smart\_Junction, starvationMaxValue*)

**Input:** a smart junction with vehicle queues.

**Output:** the conflict side that gets green light.

1. maxBenefit <- 0
2. chosen <- 0
3. maxStarve <- -1
4. For each conflict side CS<sub>i</sub> where  $0 < i \leq 4$  in *Smart\_Junction* check if CS<sub>i</sub> is the most starved:
  - 4.1 If  $css[i] \geq starvationMaxValue$ 
    - 4.1.1 If  $CSS[i] > maxStarve$ 
      - 4.1.1.1 maxStarve <- CSS[i]
      - 4.1.1.2 chosen <- CS<sub>i</sub>
5. If maxStarve > -1
  - 5.1 Return chosen
6. For each conflict side CS<sub>i</sub> where  $0 < i \leq 4$  in *Smart\_Junction* check if CS<sub>i</sub> has the largest benefit:
  - 6.1 If maxBenefit < CSB[i]
    - 6.1.1 maxGrade <- CSB[i]
    - 6.1.2 Chosen <- CS<sub>i</sub>
7. Return chosen

*VCG algorithm*

Per each time stamp, determined by the length of a green light, the VCG algorithm calculates the amount of points a vehicle should pay for promoting its location.

A vehicle does not pay if the following occurs:

- Vehicle belongs to a conflict side which does not receive a green light.
- Vehicle belongs to a conflict side which receives a green light, but the vehicle does not contribute to the green light allocation. Non-contribution is defined if the vehicle's conflict side receives a green light regardless of the vehicle's benefit value (the CSB of the vehicle's conflict side is re-calculated by ignoring the vehicle's benefit value).

A vehicle will pay if it contributes to the allocation of a green light to its conflict side. First, the CSB of the vehicle's conflict side is re-calculated using the benefits of all vehicles belongs the conflict side, except for the benefit of the vehicle itself. Second, a new conflict side with the maximal CSB time is selected. Third, the amount of points paid is the difference between the new maximal CSB time and the newly calculated CSB for the vehicle conflict side.

**VCG**(*Smart\_Junction, V, chosenCS*)

**Input:** a smart junction with vehicle queues, vehicle, the vehicle conflict side.

**Output:** the payment the vehicle has to pay.

1. Payment <- 0
2. benefitWithoutV <- CSB[chosen] – the benefit of V
3. biggest <- -1
4. for each conflict side CS<sub>i</sub> where  $0 < i \leq 4$  in *Smart\_Junction*
  - 4.1 if CS<sub>i</sub> ≠ chosenCS
    - 4.1.1 if CSB[i] > benefitWithoutV
      - 4.1.1.1 then biggest <- i
5. if biggest ≠ -1
  - 5.1 then payment <- CSB[biggest] – gradeWithoutV
6. Return payment

For the algorithm we take  $n$  as the total number of vehicles in the junction. Steps 1-3 are initializations of the variables of the payment, the benefit of the chosen conflict side if vehicle  $V$  was removed from it, the  $i$  value of the conflict side that would be chosen if  $V$  did not exist. The complexity of these steps is  $O(1)$ . Step 4 goes iteratively over all the conflict sides, which we know are four. Therefore, the complexity of this step is  $O(1)$  as well. Step 5 updates the payment according to the algorithm findings. Step 6 returns the payment the vehicle  $V$  has to pay. Both of these steps' complexity is  $O(1)$ . Hence, the total complexity of the algorithm is  $O(1)$ .

### Junction Evacuation

This is the main junction evacuation algorithm, which operates until all vehicles are removed from the junction. Each time a new green light should be allocated, a conflict side is selected, and a green light is operated for a certain amount of time. Green light timing is determined by the number of vehicles waiting within the conflict side queue up to the maximal green light time (set to 60 seconds). During green light time, each vehicle moves towards the intersection, where some of the vehicles pass the junction and are removed from the conflict side queue. Each vehicle that crosses the junctions pays some points, calculated based on the VCG algorithm.

#### **ClearJunctionByVehiclesAttributesPrioritiesAndBenefit(Smart\_Junction, Max\_Green\_time)**

**Input:** a smart junction with vehicle queues, maximal green light time per conflict side.

1. vCross: vector that contains the vehicles that pass the junction during the green light.
2. Chosen  $\leftarrow$  -1; Payment  $\leftarrow$  0; greenLightCounter  $\leftarrow$  0;
5. for each conflict side CS<sub>i</sub> where  $0 < i \leq 4$  in *Smart\_Junction* initialize the starvation parameter to 0:
  - 5.1 CSS[i]  $\leftarrow$  0
6. while there are still vehicles in the *Smart\_Junction*:
  - 6.1 for each conflict side CS<sub>i</sub> where  $0 < i \leq 4$  in *Smart\_Junction* calculate the benefit and time to cross:
    - 6.1.1 For each Direction D<sub>i, j</sub> where  $1 \leq j \leq 4$  (maximal 4 directions in conflict side) in CS<sub>i</sub> do:
      - 6.1.1.1 For each lane L<sub>i, j, k</sub> where  $1 \leq k \leq |D_{i, j}|$  (number of lanes in direction) do:
        - 6.1.1.1.1 For each vehicle V<sub>i, j, k, l</sub> where  $1 \leq l \leq |L_{i, j, k}|$  (number of vehicles in lane) do:
          - 6.1.1.1.1.1 CSB[i]  $\leftarrow$  CSB[i] + V<sub>i, j, k, l</sub> benefit
          - 6.1.1.1.1.2 CST[i]  $\leftarrow$  CST[i] + V<sub>i, j, k, l</sub> cross time
          - 6.1.1.1.1.3 If CST[i] > Max\_Green\_time
            - 6.1.1.1.1.3.1 Then CST[i]  $\leftarrow$  Max\_Green\_time
  - 6.2 greenLightCounter  $\leftarrow$  greenLightCounter +1
  - 6.3 chosen  $\leftarrow$  OpenGreenLight(*Smart\_Junction*, starvationMaxValue)
  - 6.4 vCross  $\leftarrow$  the vehicle from *Smart\_Junction* that crosses in this green light from the chosen conflict side.
  - 6.5 for each vehicle V<sub>i</sub> in vCross calculate payment, enter V<sub>i</sub> data to log file and remove V<sub>i</sub> from *Smart\_Junction*:
    - 6.5.1 payment  $\leftarrow$  VCG(*Smart\_Junction*, V<sub>i</sub>, chosen)
    - 6.5.2 enter vehicle data to log file
    - 6.5.3 remove V<sub>i</sub> from the chosen conflict side
  - 6.6 for each conflict side CS<sub>i</sub> where  $0 < i \leq 4$  in *Smart\_Junction* update the starvation counter:
    - 6.6.1 if CS<sub>i</sub>  $\neq$  chosen
      - 6.6.1.1 CSS[i]  $\leftarrow$  CSS[i] +1
    - 6.6.2 Else CSS[i]  $\leftarrow$  0
7. Write all the junction data to the log file of the junction.

For the algorithm we take  $n$  as the total number of vehicles in the junction. Steps 1-4 are initializations of the variables of the vehicles' vector that crossed the junction, the chosen conflict side, payment for a vehicle in the junction, the counter of the amount of green lights. The complexity of these steps is  $O(1)$ . Step 5 goes iteratively over all the conflict sides, which we know are four, and initializes the starvation parameter. Therefore, the complexity of this step is  $O(1)$  as well. Step 6.1 goes iteratively over all the conflict sides, which we know are four, and calculates their benefit sum and the time they need the green light to last. This step runs over all the vehicles in the junction, and therefore, its complexity is  $O(n)$ . Steps 6.2 and 6.3 update the counter of green lights and decide which conflict side gets to pass in this green light. Therefore, their complexity is  $O(1)$ . Steps 6.4 and 6.5 group the vehicles that passed the junction, calculate their payment and remove them from the junction. Therefore, their complexity is  $O(n)$  on the upper end. Step 6.6 goes iteratively over all the conflict sides, which we know

are four, and updates the starvation parameter. Therefore, the complexity of this step is  $O(1)$ . The conclusion from this analysis is that the complexity of step 6 is  $O(n)$ . Step 7's complexity is  $O(1)$ . Hence, the total complexity of the algorithm is  $O(n)$ .

### 3. Results

The algorithm suggested in this study uses a vehicle social priority function with a personal benefit factor in order to promote vehicles' status while waiting at a junction. A negative relation is expected between a social priority value and a benefit value regarding the junction crossing time. Higher social priority and benefit value is expected to correlate with a shorter junction crossing waiting time. This is true in the case where each conflict side contains a single vehicle. In this simulation, each conflict side contains many vehicles, each of them has a different social priority and benefit value. The junction promotion of a single vehicle is influenced by the social priority and benefit values of all the other vehicles located within the vehicle conflict side. Due to the mutual vehicle influence, we define a new goal which promotes every vehicle crossing time to its optimal crossing time. In order to define the optimal crossing time for each vehicle, each of the two variables, social priority and personal benefit, was grouped into 3 value ranges, where each range contains a similar number of vehicles.

Table 1 presents the number of vehicles by social variable (priority, benefit) and value range. Since the social priority value was uniformly distributed among vehicles, each social preference category contains one third of the variable scale. The personal benefit values distribution was biased towards the lower values. In order to compensate for this bias, we set a larger width for ranges that had higher values.

**Table 1.** Number of vehicles by social variable and value category

Value ► variable ▼	High	Medium	Low
Priority	$x \leq 3$ 526	$3 < x \leq 6$ 545	$x > 6$ 526
Benefit	$x \leq 25$ 578	$25 < x \leq 70$ 586	$x > 70$ 436

Table 2 presents the expected optimal behavior by benefit and social priority. This division established a baseline against which we can evaluate the success of a smart junction with a social priority and benefit algorithm, as compared to a regular junction (junction which sets a fixed green light order with predefined green time length). Table 2 presents five types of expected optimal behavior, indicated by numbers enclosed in the brackets. Vehicles with high benefit and priority values are expected to be the first to cross the junction. At the other end, vehicles with low benefit and priority values are expected to be the last to cross the junction. Tables 3 and 4 presents the number of vehicles and expected optimal crossing time range, respectively, by benefit and priority.

**Table 2.** Expected optimal behavior by benefit and priority

Benefit ► Priority ▼	High	Medium	Low
High	Reduce time (1)	Reduce time / no change (2)	Any/no change (3)
Medium	Reduce time / no change (2)	Any/no change (3)	Increase time / no change (4)
Low	Any/no change (3)	Increase time / no change (4)	Increase time (5)

**Table 3.** Number of vehicles by benefit and priority

Benefit ► Priority ▼	High	Medium	Low
High	296	153	80
Medium	140	261	144
Low	0	172	354

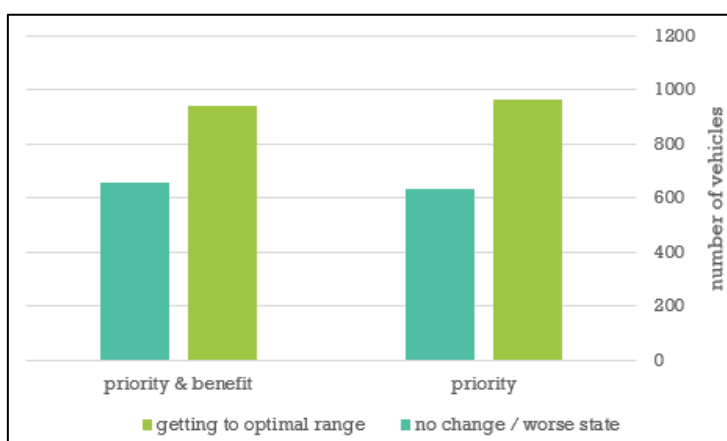
**Table 4.** Expected optimal junction crossing time range by benefit and priority

Benefit ►	High	Medium	Low
Priority ▼			
High	1-1814	1593-3517	3408-5563
Medium	1593-3517	3408-5563	5401-7294
Low	3408-5563	5401-7294	7188-9223

The expected optimal junction crossing time was calculated based on the expected optimal behavior, presented in Table 2, the number of vehicles presented in Table 3, and the simulation results. For each type of optimal behaviour, as defined in Table 2, we calculated the range of traffic lights required for vehicles to cross the junction. The range was calculated based on the number of vehicles belonging to the behavior type (presented in Table 3). The traffic lights range is translated into a time range. Since a range of traffic lights can pass more vehicles than defined per each type, an overlap of one traffic light between ranges was taken.

Here are two examples to clarify expected crossing time range calculation.

The 296 vehicles belonging to the high priority and benefit values (indicated by 1 in Table 2) are expected to be the first to cross the junction. Based on the simulation results, it takes 1-9 traffic lights, translated to 1-1814 seconds, for the first 296 vehicles to cross the junction. The next group, indicated by 2 in Table 2, contains 293 vehicles (153+140). It takes 9-18 traffic lights (the 9th traffic light is overlapped with the previous range), translated to 1593-3517 seconds for these to cross the junction. Figure 2 presents the number of vehicles in the smart social junctions (priority and priority & benefit junctions) which were closer to their optimal behaviour, as compared to a regular junction. As described in Tables 2-4, optimal behaviour is defined by expected crossing junction time range. A vehicle with certain priority and benefit values that crossed the junction within the expected crossing range is counted as a vehicle which "approached its optimal range". In addition, a vehicle with a position closer to the optimal range, as compared to a regular junction, is also counted as a vehicle which "approached its optimal range". Here are two examples to clarify this calculation: (i) The optimal range for the reduced time is 1-1814 seconds. A vehicle has a crossing time of 1900 seconds at the smart social junction and 2000 at the regular junction. This vehicle is counted as "approached its optimal range". (ii) The optimal range for increased time is 7188-9223 seconds. A vehicle has a crossing time of 7000 seconds at the smart social junction and 6500 at the regular junction. This vehicle is counted as "approached its optimal range".



*Figure 2.* Proximity to optimal behavior in a smart social junction, as compared to a regular junction

Results show that at the smart social junctions (priority & benefit and priority alone) 60% of the vehicles were closer to their "optimal behaviour", as compared to the regular junction. Figure 3 presents the number of vehicles in the priority & benefit junctions which were closer to their optimal behaviour, as compared to a priority junction. Results show that 65% of the vehicles in the priority & benefit junction improved their crossing time and approached their optimal behaviour, as compared to their crossing time in the priority junction.

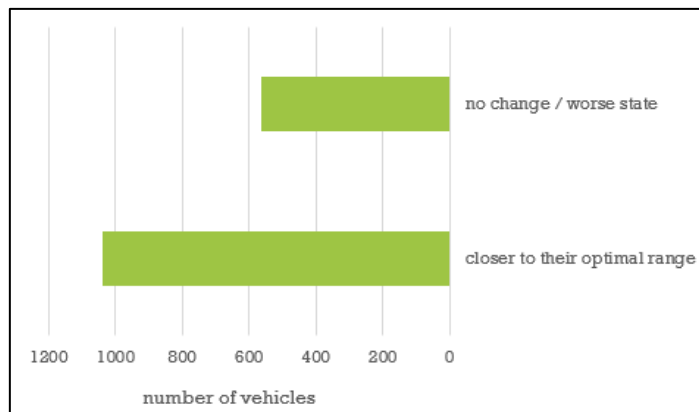


Figure 3. Proximity to optimal behavior in priority & benefit junction as compared to priority junction

Figure 4 presents the trade-off between the vehicle's properties and their proximity to its "optimal behaviour". Each smart junction vehicle is represented by a point, while the X axis is the vehicle's priority and the Y axis is the vehicle's benefit. A green point represents a vehicle that is closer to its "optimal behaviour", while a blue point represents a vehicle that has the same state or got worse. As noted, the benefit is calculated by multiplying a random value extracted from a range of 0-20% with the vehicle's total amount of points determined by the priority. Consequently, the benefit range increases as the priority increases, and each priority value contains approximately the same number of vehicles, but the vehicles' benefit values are more scattered as the priority increases.

As the priority increases the number of green vehicles increases, indicating that more vehicles approach their "optimal behaviour". In addition, as the priority decreases the vehicle's driver should invest more benefit points in order to improve his state. For example, for priority value equals to 4, the green points (vehicles) are the ones with the higher values of benefit points. For a priority value that is equal to 9, the green vehicles are spread all over the range of benefit points. It can be said that as the priority value increases the contribution of benefit points to promoting the vehicle's junction position is less significant.

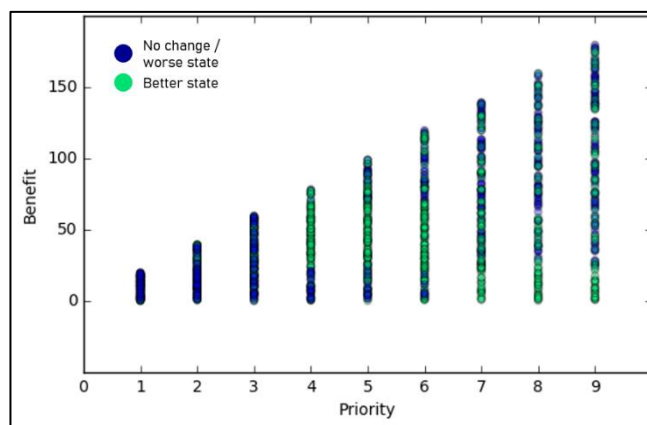


Figure 4. Junction vehicles status across social priority and benefit

#### 4. Conclusions and future work

Previous research (Barzilay *et al.*, 2018a,b) has presented an algorithm that controls traffic light timings by considering social priorities. This paper incorporates a VCG auction protocol into the smart social junction algorithm in order to provide a mechanism to activate the social priority when required. The benefit from promoting one's position in the junction queue is used for the auction protocol. Results show that the VCG auction protocol based on both benefit and social priority enables better overall junction navigation matching the vehicle's unique characteristics, that is, its social priority. It can be argued that the mechanism we proposed provides the intersection with characteristics of artificial intelligence that enables personalization of the vehicle.



In this paper we restricted some of the junction physical features, such as the vehicles' speed and length, and extracted the emergency vehicles.

Further work should investigate the VCG auction protocol with more realistic junction features, manipulating the frequency of rescue cars and the social values given to vehicles. Another enhancement could be to examine the impact of different traffic load values on the auction protocol efficiency.

## References

1. Khalid, A.S., Al-Khateeb, Jaiz A.Y., Johari, Wajdi F., Al-Khateeb (2008). Dynamic Traffic Light Sequence, *Science Publications. Journal of Computer Science*, 4 (7), 517–524. DOI:10.3844/jcssp.2008.517.524.
2. Collotta, M., Conti, V., Scatà, G., Pau, G., & Vitabile, S. (2013). Smart wireless sensor networks and biometric authentication for real time traffic light junction's management. *International Journal of Intelligent Information and Database Systems* 6, 7(5), 454-478.
3. Pau, G., & Scata, G. (2014). Smart traffic light junction management using wireless sensor networks. *WSEAS transactions on communication*, 14, 2224-2864.
4. Younes, M. B., Boukerche, A. (2017). An efficient dynamic traffic light scheduling algorithm considering emergency vehicles for intelligent transportation systems. *Wireless Networks*, 1-13.
5. Veldscholten, N. (2015). Moral reasoning in traffic: About the possible relations between moral reasoning and traffic safety (*Master's thesis, University of Twente*).
6. Reynolds, S.J. & Ceranic, T.L. (2007). The effects of Moral Judgment and Moral Identity on Moral Behavior: An Empirical Examination of the Moral Individual, *Journal of Applied Psychology*, 92 (6), 1610-1624. DOI: 10.1037/0021-9010.92.6.1610
7. Barzilay, O., Voloch, N., Hasgall, A., Lavi Steiner, O., Ahituv, N. (2018a). Traffic Control in a Smart Intersection by an Algorithm with Social Priorities. *Contemporary Engineering Sciences*, 11(31), 2018, 1499–1511.
8. Barzilay, O., Voloch, N., Hasgall, A., Lavi Steiner, O. (2018b) Real life applicative timing algorithm for a smart junction with social priorities and multiple parameters, *2018 IEEE International Conference on the Science of Electrical Engineering in Israel (ICSEE)*, Eilat, Israel, 2018, 1-5. DOI:10.1109/ICSEE.2018.8646018
9. Groves, T. (1973). Incentives in Teams, *Econometrica*, 41 (4): 617–631.
10. Carlinom, D., Boyles, S.D., Stone, P. (2013) Auction-based autonomous intersection management, *16th Intelligent Transportation Systems Conference*, Hague, Netherlands, 2013, DOI:10.1109/ITSC.2013.6728285