Proactive Control of Traffic in Smart Cities

Benjamín Zavala and Germán H. Alférez

Global Software Lab,
School of Engineering and Technology,
Montemorelos University, Mexico

Traffic Congestion in Big Cities

- In recent years, the world has experienced **mass migration** from the countryside to the city.

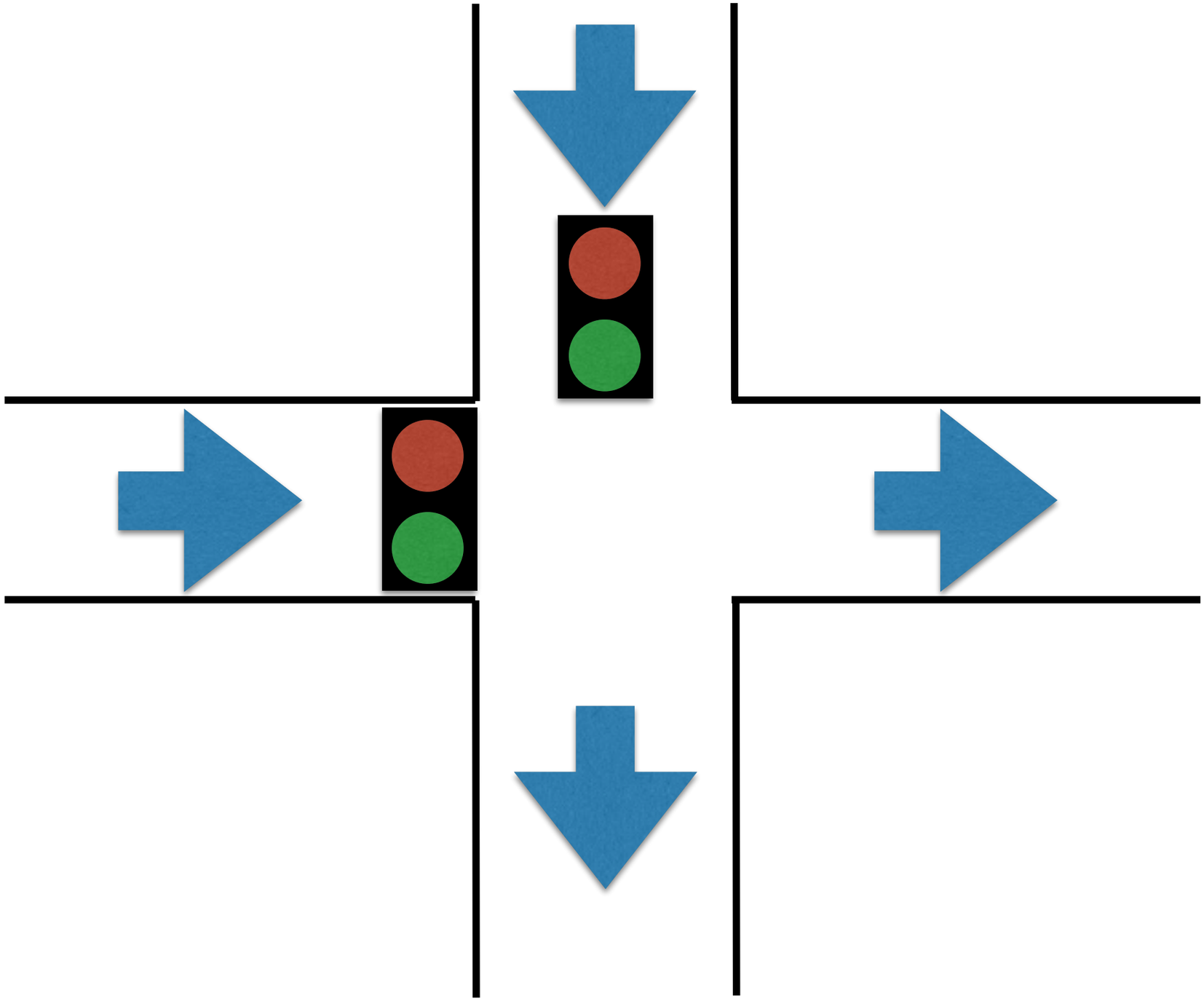
- For example, in Mexico today over **70%** of the population lives in the **city** [1].

International Business Times

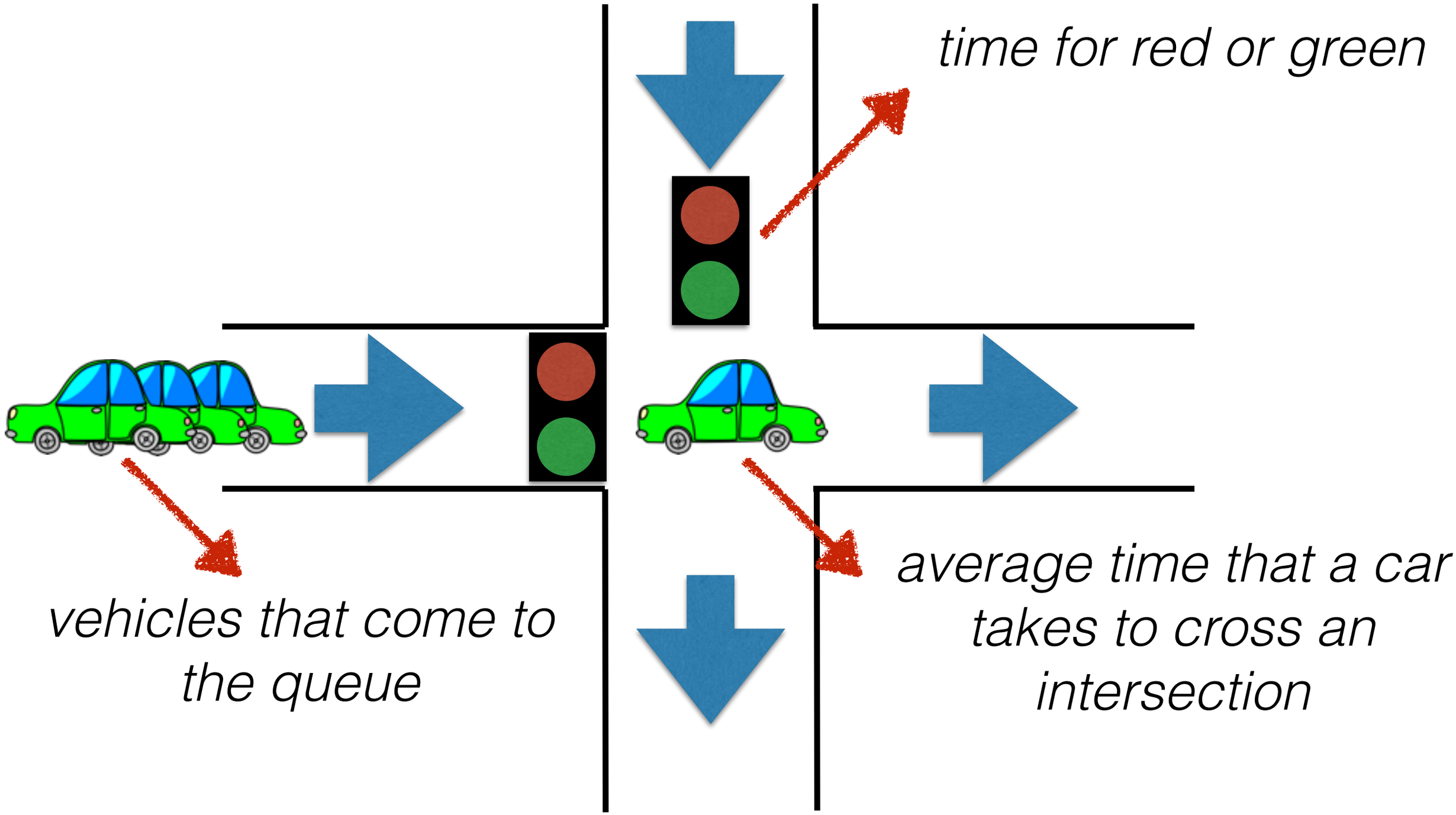
Traffic Congestion in Big Cities (Cont.)

- This phenomenon leads to many problems.
  - **Traffic congestions negatively affect the quality of life of citizens by increasing:**
    - Travel time
    - Stress
    - Economic lost
    - Environmental pollution

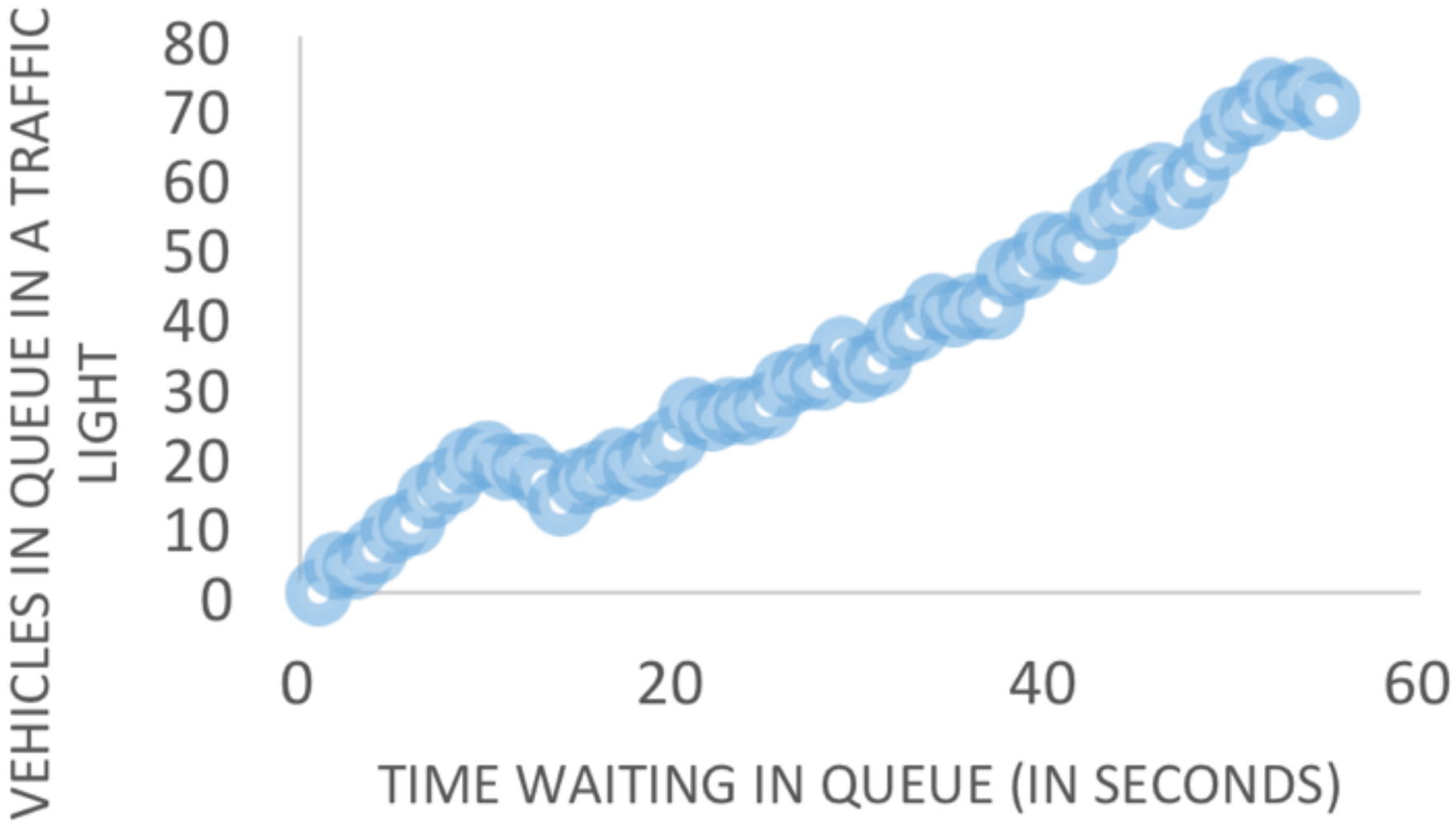
Traffic Congestion in Big Cities (Cont.)



Traffic Congestion in Big Cities (Cont.)



Traffic Congestion in Big Cities (Cont.)



Time for red or green =15 seconds
Vehicles that come to the queue = 0 - 9 cars
Average crossing time = 3 seconds

Reactive Vs. Proactive

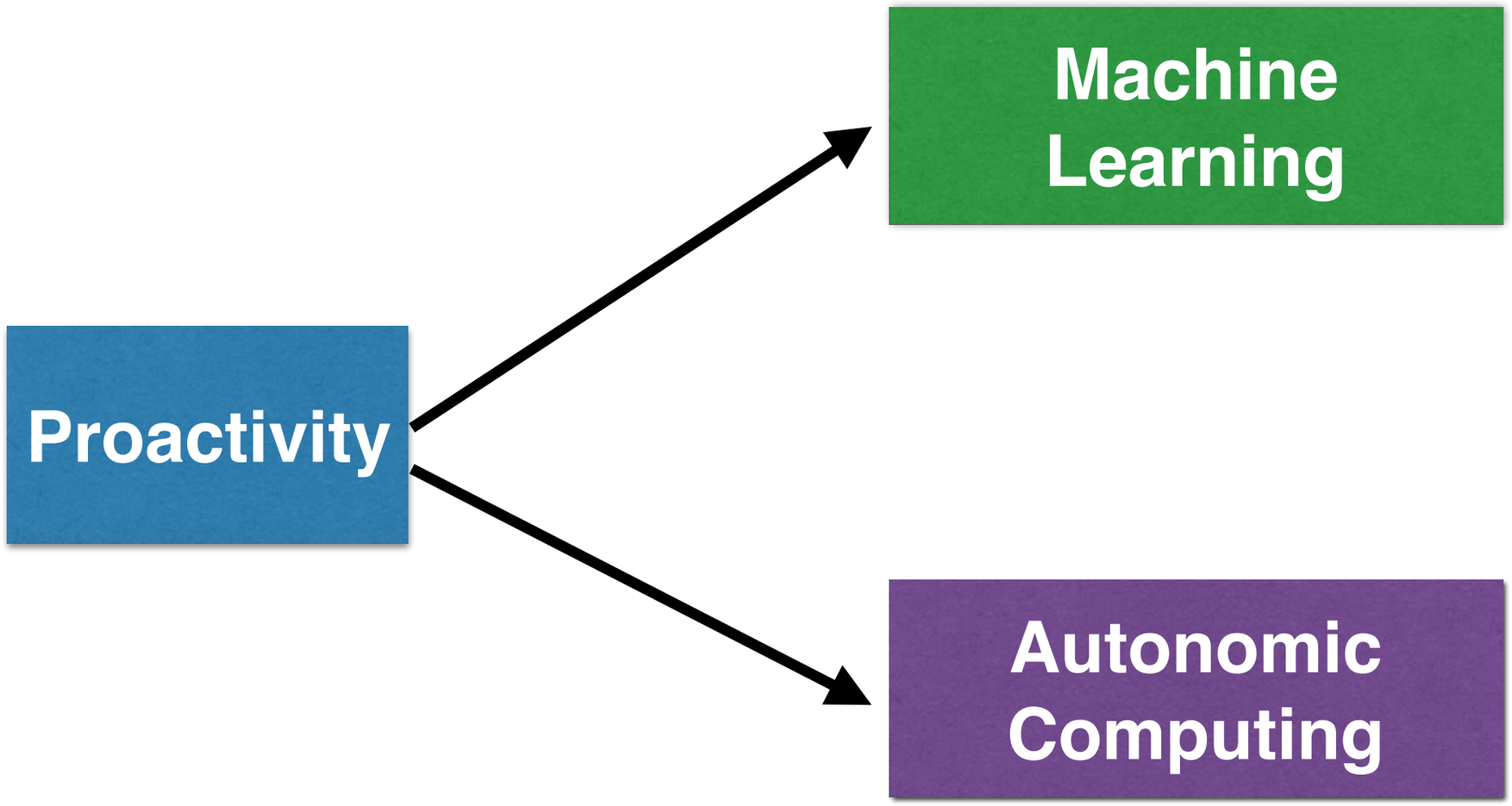
Reactive solutions to control traffic in smart cities wait for an **event to happen** (i.e., a long queue at a traffic light) to generate an action to try to solve this event.

Examiner

Reactive Vs. Proactive (Cont.)

For **cities** to become truly **intelligent**, they need "**proactive**" solutions that **anticipate traffic problems** and **prevent these problems from becoming evident**.

Basis of Our Solution



Basis of Our Solution (Cont.)

Machine Learning

Forecasting is the process of making statements about events whose actual outcomes have not yet been observed [9].

Basis of Our Solution (Cont.)

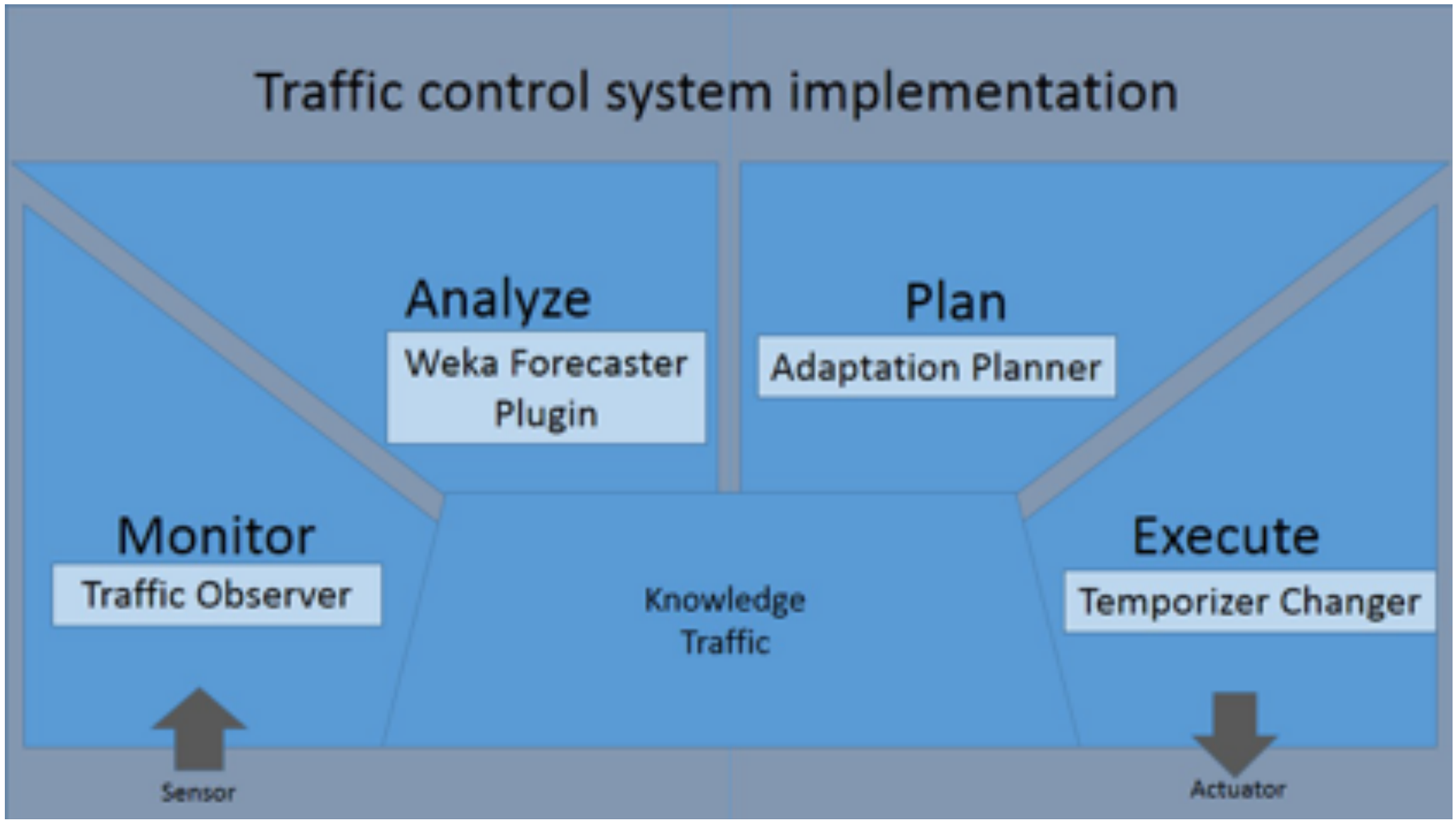
Autonomic Computing

- The AC initiative is inspired by the **human body's autonomic nervous system** (monitors **unconsciously** heartbeat, blood sugar level, body temperature).

- It has evolved as a discipline to create **self-managing software** to overcome the complexities to maintain systems effectively.

- **IBM's reference model for autonomic control loops** (which is sometimes called the **MAPE-K loop**) [10].

Our Solution

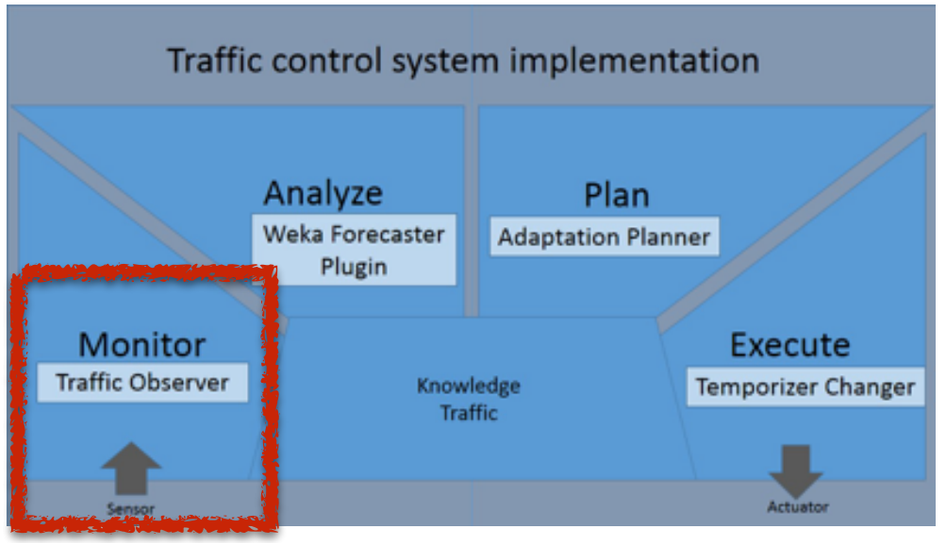


Monitor

- Monitoring involves **capturing properties** of the **context** that are meaningful to the self-properties of the system.

- **Traffic Observer:** observes the traffic through **sensors**.

  - The Traffic Observer periodically checks the activity in the traffic simulation.



Monitor (Cont.)

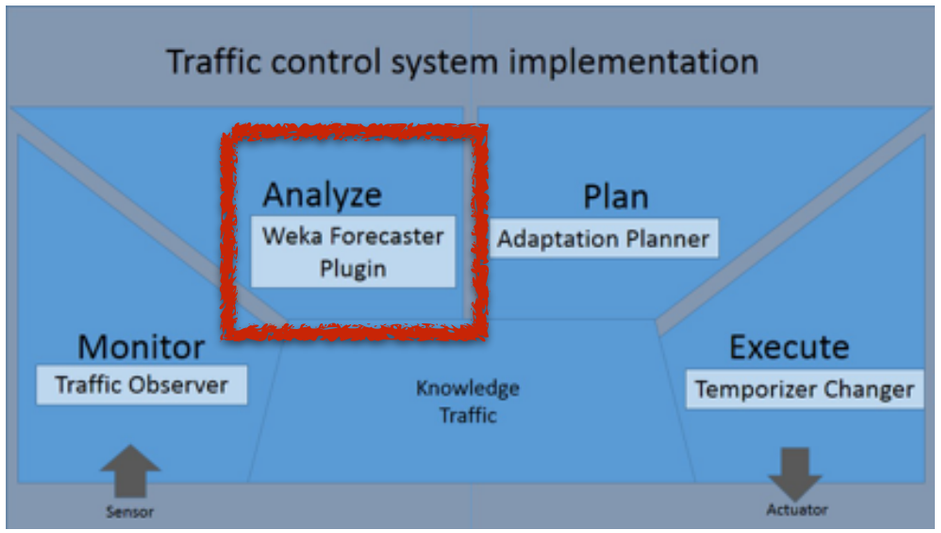
- The **Traffic Observer** detects if the **number of vehicles** queued at a certain moment at the traffic light **is greater than the Service Level Agreement (SLA)**, i.e., a violation of the SLA.

- If so, this event is saved in a log file with **.arff extension**.

@relation A @attribute seconds numeric
@attribute cars numeric @data 60, 7

Analysis

The objective of this phase is to **detect**, in a **proactive** manner, the **traffic problems** that may occur according to the results captured in the log generated by the Traffic Observer.



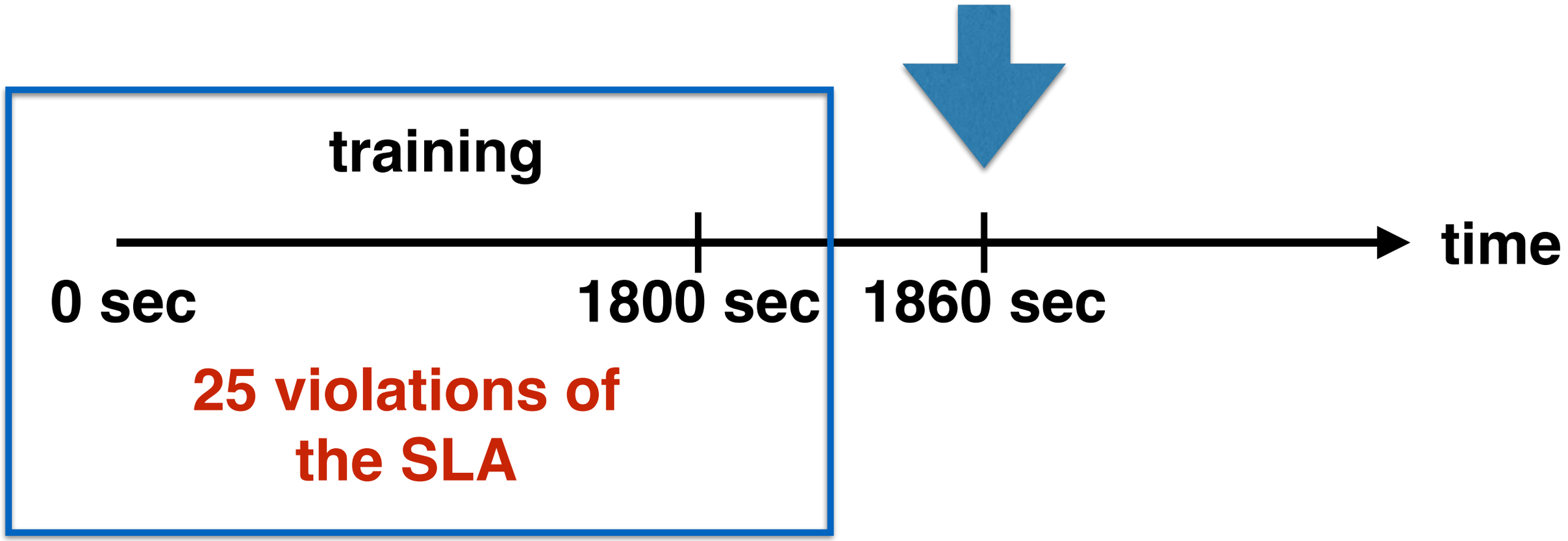
Analysis (Cont.)

- In order to accomplish the prediction of traffic problems, we use the **Weka Forecasting plugin.**
  - This plugin can load or import a time series forecasting model and use it to generate a forecast for future time steps beyond the end of incoming historical data [13].

- **Multilayer Perceptron.**

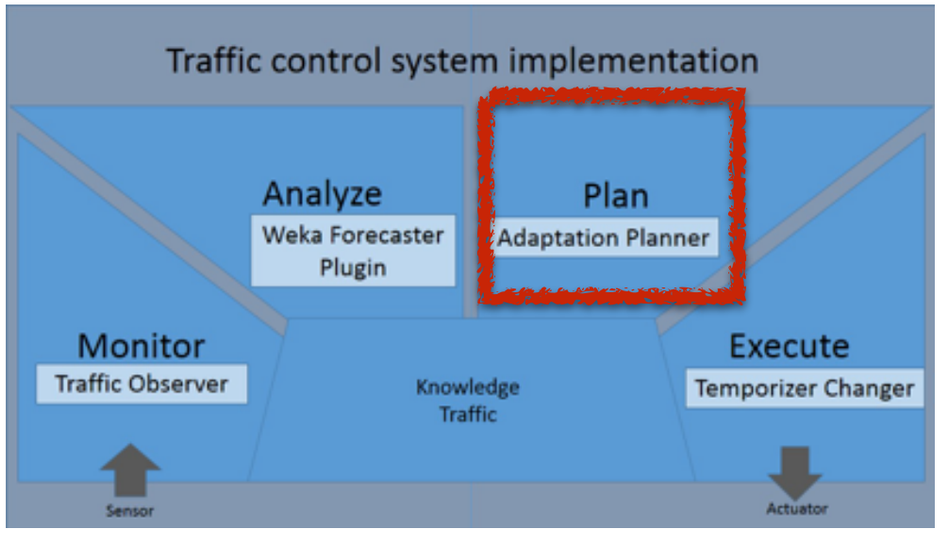
Analysis (Cont.)

The forecaster predicts SLA violation: 35 cars in queue



Planning

- The objective of this phase is **to plan how to automatically solve the traffic problems** predicted in the analysis phase.



- **Adaptation Planner:** plans changes in **traffic light timers**.

Planning (Cont.)

1. The **Adaptation Planner** keeps in a variable the text recovered from the file generated in the previous phase.

    For example, the **Adaptation Planner** takes the data of 35 vehicles in queue that will violate the SLA at some future time (after training) according to forecasting.

Planning (Cont.)

2. The **Adaptation Planner** performs the following operation:

newTimer = (int) (d * tCross)

d = numbers of cars that could violate the SLA according to forecasting

tCross = average time that a car takes to cross an intersection

The value of the "*newTimer*" variable, which corresponds to the solution to a problem (i.e., a possible violation of the SLA), is saved.

Planning (Cont.)

For example,

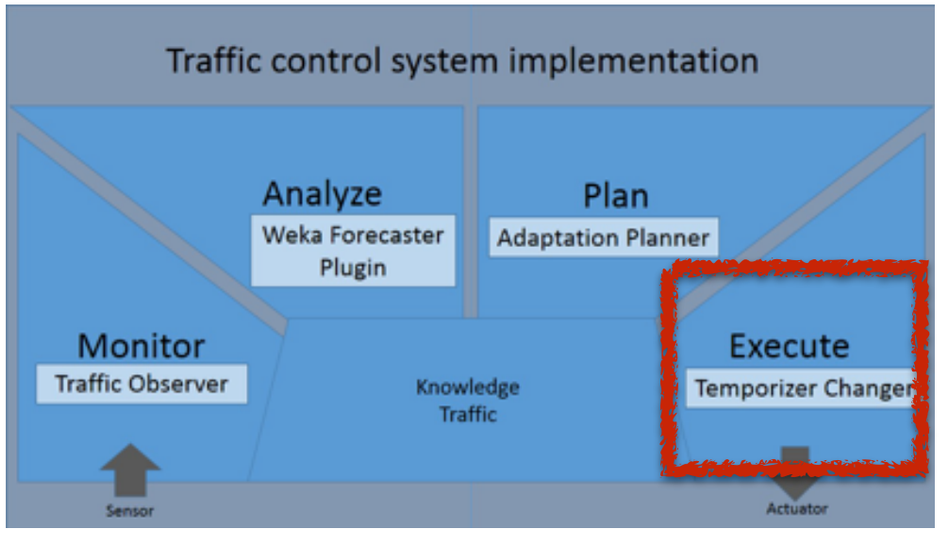
- A vehicle takes around **3 seconds** to cross the intersection (***tCross***)
- There are **35 expected vehicles** that will arrive at the traffic light (***d***).

The result of ***newTimer*** is **105 seconds** *(newTimer = (int) (d * tCross))*

It means that 35 vehicles take 105 seconds to cross the intersection.

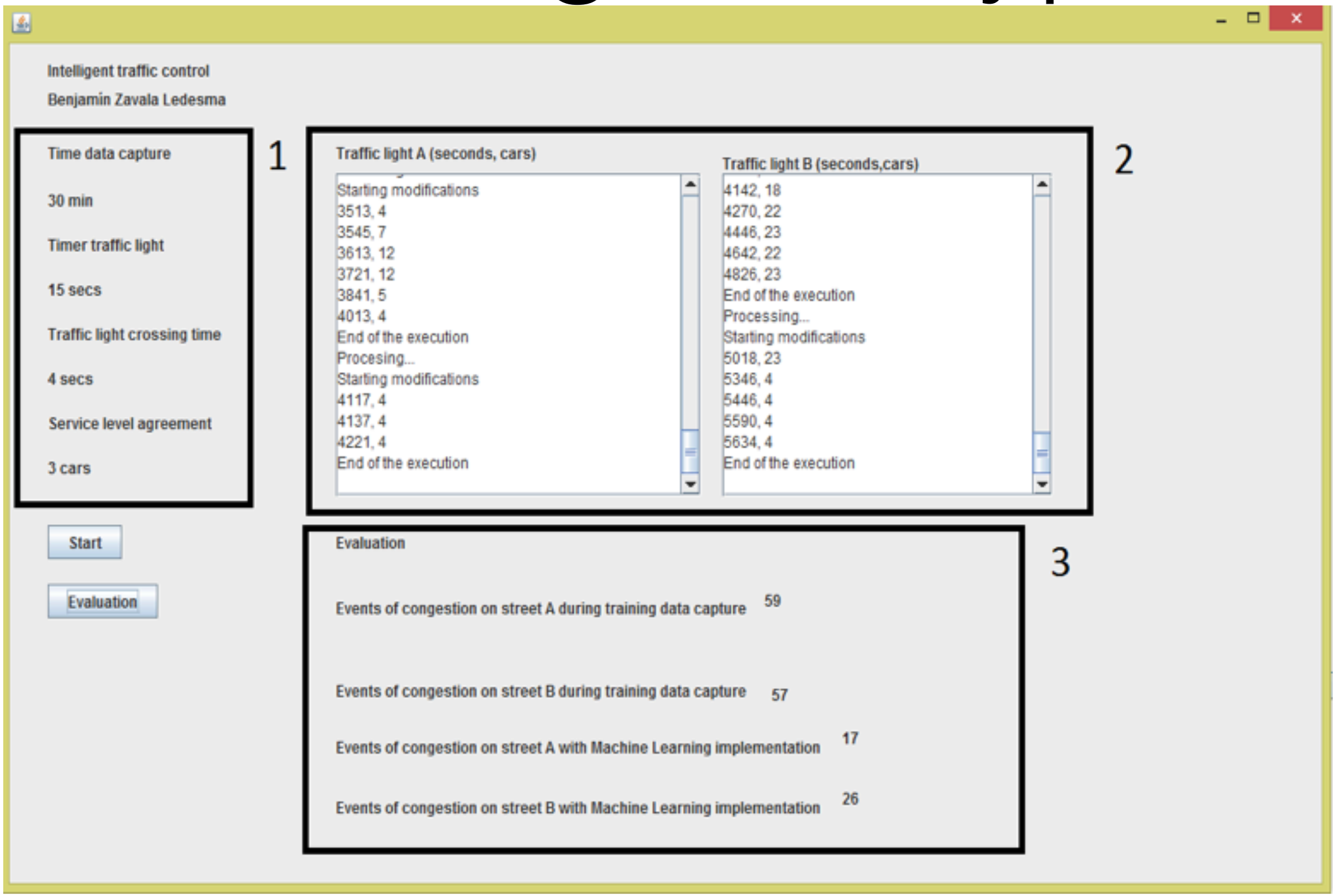
Execution

- The objective of this phase consists of making **changes** in the timers of the **traffic lights** according to the results of the planning phase.



- **Actuators** are in charge of making changes in traffic light temporizers (according to "*newTimer*").

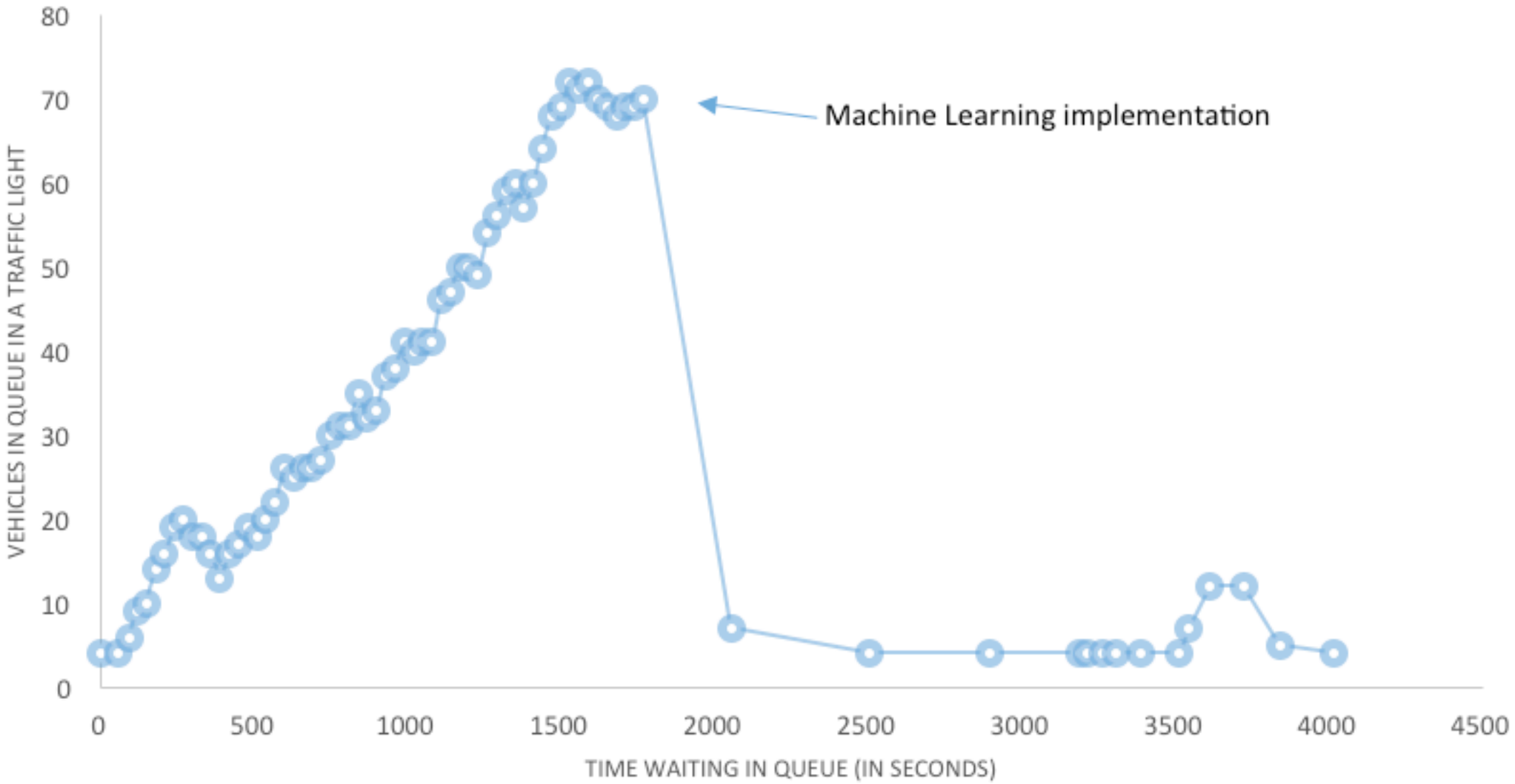
Running Prototype



Running Prototype (Cont.)

- During training, the number of times that the **SLA was violated** is **pretty high** at each traffic light:
  - 59 violations at the traffic light A
  - 57 violations at the traffic light B
- **By using forecasting the number of SLA violations descended dramatically**
  - 17 violations at the traffic light A
  - 26 violations in traffic light B

Running Prototype (Cont.)



Conclusions and Future Work

- A **proactive** solution to traffic control by means of **Machine Learning** and **Autonomic Computing**.

- As future work, we will implement a **computer vision module** for live traffic control through cameras.

  - The Traffic Observer will collect this data.

- We will also develop a **mobile application** to help users make queries about the status of the traffic.

Thank you!