# Green Time Split Optimization by Dynamic Programming

## Master‘s Thesis of Tom Bagehorn

**Mentoring:**
M.Sc. Eftychios Papapanagiotou
Dipl.-Ing. Jakob Kaths

**External Mentoring:**
Prof. Dr.-Ing. Andreas Poschinger (Hochschule München)

### Motivation

The quality of urban intersections mainly depends on the capacity of its intersections and their generated delay. Therefore, many urban intersections are controlled by traffic signals to improve the traffic quality and safety while reducing pollutant air emissions.

In this thesis steps are made toward a new method for signal timing optimization, based on backward dynamic programming which is applicable under real-world conditions and which is enabled not only for public transport prioritization but also for cooperative systems.

### Dynamic Programming

Dynamic programming is able to solve multistage decision problems by calculating the optimal series of single decision steps backward in time as shown by the blue arrow in the top part of **Fig. 1**.

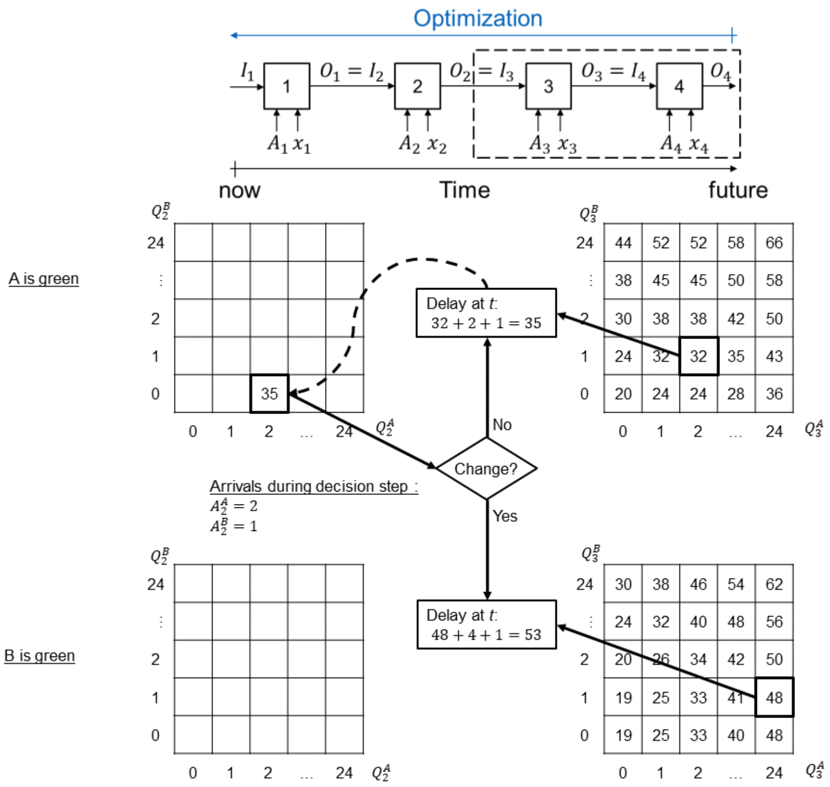


**Fig. 1** Principles of Dynamic Programming in Split Optimization

A split optimization using dynamic programming is Robertson and Bretherton’s DYPIC. The bottom part of Fig.1 shows one optimization step of one queuing state (0 vehicles for signal B and 2 for signal A): signal A is continued since the total delay of this decision is 35 compared to 53 if the signal would be changed to B. One optimization step in DYPIC uses a time step width of five seconds and a service rate of 2 vehicles / 5 seconds. Together with the arrival rates (in **Fig. 1**: 2 vehicles for signal A and 1 for signal B), the service rate and the control decision relate the cells in the queuing state matrices. Inter-green and minimum green times in DYPIC are fixed by the optimization step width of five seconds.

### Developed Dynamic Programming Algorithms

In this thesis the ideas of DYPIC have been extended to model a more realistic service rate and to consider more arbitrary inter-green and minimum green times. This is done by using a two seconds optimization step width and allowing inter-green and minimum green times to be a multitude of those two seconds by introduction of additional control states as shown in **Fig. 2**:

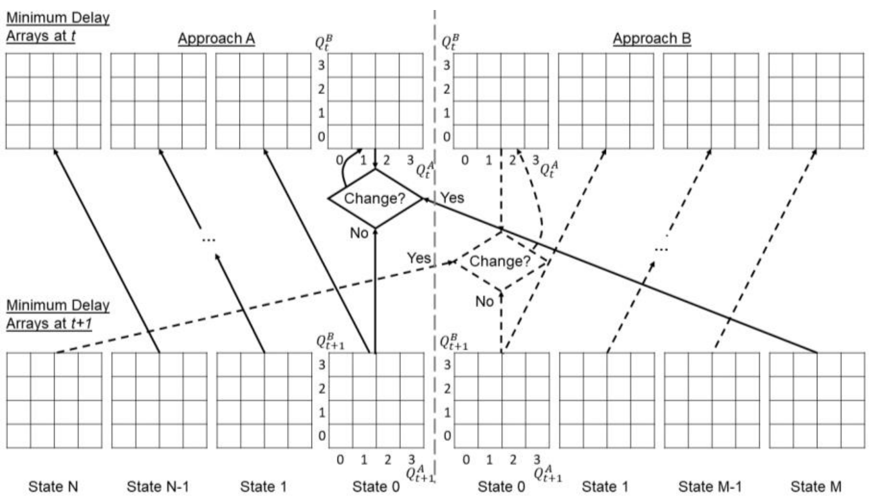


**Fig. 2** Consideration of Inter-Green and Minimum Green Times by States

Signal changes are only permitted from state 0. All other state changes are between the neighboring states and represent either inter-green or minimum green times. The algorithm also has been extended for real-time traffic control by utilizing a rolling horizon approach and vehicle data from 150m detectors.

### Evaluation

The algorithms have been evaluated in simulations. A two-stage intersection with two arriving and departing legs has been simulated using real inter-green times and realistic service rates (see **Fig. 3**).

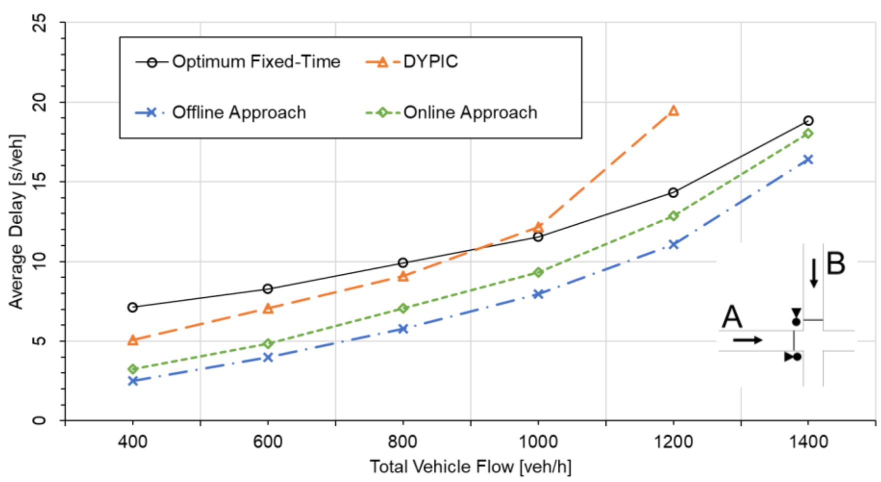


**Fig. 3** Performance of the Proposed Algorithms

The proposed offline and online algorithms are superior to DYPIC and an optimum fixed-time control. The optimum fixed time split is calculated for each traffic demand individually using the formulas of the RiLSA. A small but very complicated real-world intersection exceeds the capabilities of the online algorithm. The arising size of memory and the computation power needed for the calculation of the entire state space makes a simplification of the optimization model necessary. However, with the made simplifications the performance of the new method does not succeed the performance of an optimum fixed time split.

In theory dynamic programming has an outstanding performance, but due to the high complexity of real-world intersections, simplifications must be made which make dynamic programming impractical.