SYNTACTIC METHOD FOR VEHICLES MOVEMENT DESCRIPTION AND ANALYSIS

Jan Piecha, Marcin Staniek
Faculty of Transport
Silesian University of Technologies
Krasińskiego 8, 40-019 Katowice, Poland
{jan.piecha, marcin.staniek}@polsl.pl

Abstract. *The syntactic primitives and the description language can be used for assignment and analysis of vehicles movement. The paper introduces a method that allows the user spotting on an image, registered by video camera, the manoeuvres of cars on and between traffic lanes. The algorithms of these vehicles movement trajectories analysis is considered in this paper.*

Keywords. Syntactic primitives, vehicles' trajectories, description language, manoeuvres of moving cars.

1 Introduction

The video-camera usage for a traffic data on-line (real time) registration needs various simplifications for the calculation procedures. First, registration and traffic controlling algorithms have to be executed in a very limited time. There also are additional restrictions concerning completeness of the data, assigning the transportation process safety.

The digital cameras available on the market, are clocking the image 25 times per second, recording each car, driving 50 km/h, at 100 m passage of the road, 182 times [11]. This clock rate produces tremendous set of the input records, not needed for this slow process description.

From the other hand, the manoeuvres of passing cars, on or between the traffic lanes have to be caught in sampling intervals that can not be longer then the cars need to do these movements.

Taking into account the time the controller needs for the data set analysis, many factors of the above algorithms have to be optimised.

First, the grid of controlling intervals has to found, according to a number of traffic attendees and all safety relations between vehicles. The calculations complexity, the assumed time for calculations and the computer memory size - the machine is able to use, are defining the algorithms complexity.

The indicated demands of the transportation model description were presented by several works; among them [2, 4].

Analysing the vehicle's movement trajectories some simplifications of their descriptions was considered. The vehicle's route was divided into smaller elementary parts that allow assigning this route, instead of using the pixels bitmap descriptors.

The vehicle's shape and its movement trajectory can be described using the elementary graphical symbols, like: straight lines and curves.

The example of these trajectories was introduced in Fig.1. There was presented an output from a main road into a sub-way, by a road solution called road junction.

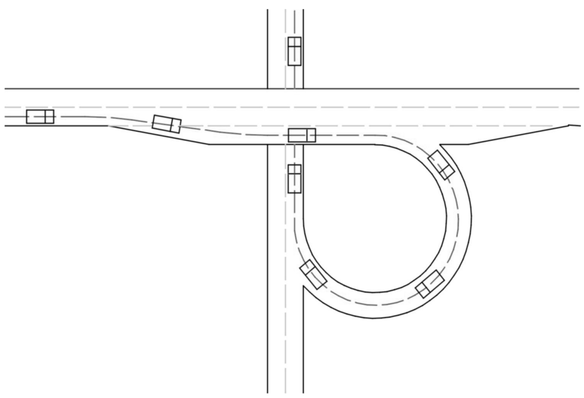

Figure 1. The example trajectory of the vehicle's movement (straight line – curve).

At the entrance of the road junction, the vehicle is changing the traffic lane, assigned by two curves and one strait line in between.

After that, the vehicle enters the curve-shape road junction. The end part of the route is a straight line.

In these trajectories description some simplifications were used. They concern the junction of the curve and the straight line descriptors. The simplifications make the algorithms less complex not loosing the quality of the description. What are more; the trajectories are defined by connections of the vehicle's layout discrete indicators, from every cell of the video image.

The needed memory size and the computing power of the traffic controller can be reduced in case we reduce the data amount into a necessary dimension.

2 The syntactic descriptors of the trajectories

In Fig.2 the illustration of the trajectory primitives are indicated. The trajectory is divided into units showing the vehicle's localisation, during its journey along a traffic lane [10].

The traffic route recognition was defined by identification processes of the vehicle's primitive descriptors, used for the assignment of the analysed objects [11, 9]. They are analysed using the description rules, of the description language.

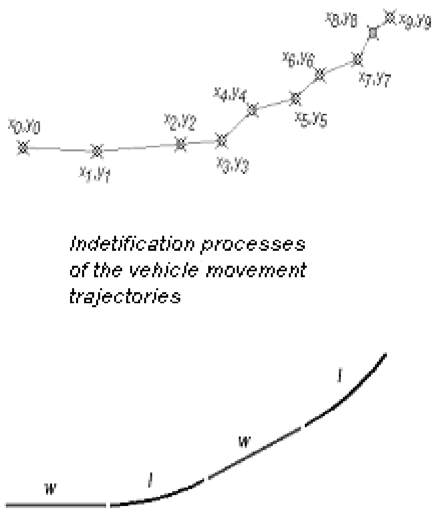

Figure 2. The syntactic primitives assignment

On a top part of the figure a geometrical coordinates were indicated. On the button part, the syntactic symbols were defined.

The trajectory description was limited by the camera observation field size and its projection onto a road surface, of the observed trajectory fragment.

The distinguished shapes of the trajectory elementary units allow us assigning the vehicle's movement by the defined syntactic symbols.

The analysis is needed for fast algorithms development, used for recognition of not eligible movements on the road.

The operation has to be executed in a real time mode of the traffic control schedule. This analysis indicates the traffic law, defining the so called traffic incidents [6].

In Fig 3 geometrical assignment of main four symbols was established: driving ahead „w", turning to the left „l", turning to the right „p" and reverse driving „c". These symbols were used for vehicle's movement trajectory description [11], [9]. These parameters are provided by an angle coordinates, like [11], [9]:

- driving ahead: w (1.8, 0.4, 0, 0.16),
- turning left: l (1.8, 0.4, 0.16, 0.16),
- turning right: p (1.8,0.4, –0.16, 0.16),
- reverse: c (1.8, 0.4, π, 0.16).

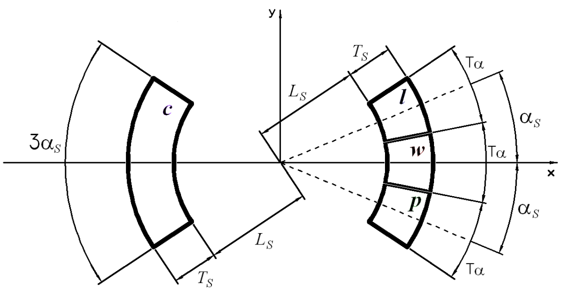

Figure 3. Geometrical assignment of the syntactic symbols

The vehicle's movement description primitives assign a present coordinates of it location, using for this purpose the primitive symbols. In case the current trajectories are close to parameters of any symbol of the description language, next fragment of the vehicle trajectory is indicated (Fig.4).

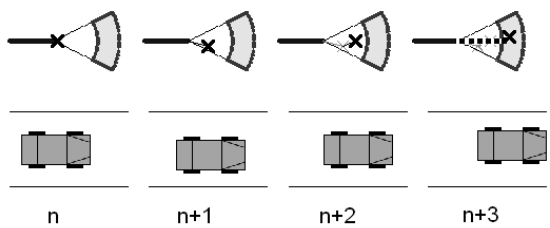

Figure 4. The vehicle's trajectory assignment in an image sequence

In subsequent video frames the vehicle's location is appointed by x. When the location vector is close to the primitives it is indicated on a light grey line (Fig.4). The dotted line is indicating replacement of the vehicle - the movement trajectory.

The illustration for the directions symbols, as: driving straight-on, turning left and right or changing the vehicle's placement, on subsequent video frames for a speed v=60km/h, were introduced in Fig.5.

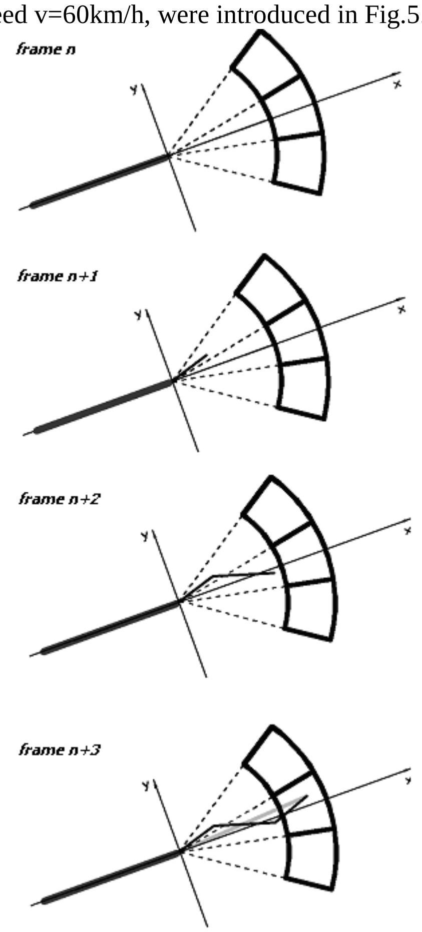

Figure 5. The vehicle's movement trajectory assignment

The real time controlling units are recently applied using gate-level technologies (PLA, FPGA, GAL, etc.). They provide us with very fast (strongly needed) image processing units. Although this technology is very fast the data size need not to be larger then it is necessary.

When the vehicle's movement description preciseness increases, the algorithms complexity is also increasing.

Simplifying the vehicles trajectories tracking procedures we reduce remarkable the calculations complexity, reducing as well its efficiency.

In traffic lanes junctions one can observe remarkable changes of vehicles speed and driving directions. High ways measurement sites are used for over-speeding vehicles monitoring. In places for parking the vehicles, movement algorithms will have different tasks for to do.

The syntactic symbols identifiers reduce the recognition difficulties, caused by different lengths or the vehicles' displacement, caused by speeds differences.

The vehicles' localisation defines their speed and the localisation history. Analysing this data we can define the vehicle's dynamic characteristics.

The recorded number of vehicle's placement, in the video sequence indicates accelerations, slowing down or braking.

The movement trajectories, observed on the traffic lane background, allows us indicate illegible manoeuvres of the vehicles (being still under analysis).

3 The trajectories description language

The discussed above symbols (primitives) for vehicles traffic description define the description alphabet $\sum$, for the introduced language; used for the traffic network assignment and modelling.

The language using the alphabet $\sum$ understands any combinations and subsets of these symbols [5]. They produce words and sentences of the language.

The symbols are used for the driving sequences description. The undefined or unrecognised movement is not classified by this recognition algorithm.

The fundamental task of the syntactic assignment method concerns the vehicle's movement and localisation recognition at the video sequence. The vehicle's movement recognition mechanism is encoded using the language grammar and generated by the grammar combination of these elementary symbols.

The fundamental definitions of the grammar [6] are expressed by relations of:

$$G = \left(\sum_N, \sum_T, P, S\right) \tag{1}$$

where:

$\sum_N$ - defines a set of non-terminals,

$\sum_T$ - set of terminals,

P - is a finite set of rules or productions,

S - is the starting symbol $S \in \sum_N$.

The alphabet $\sum$ calls the limited set of symbols, as: $\sum = \sum_N \cup \sum_T$ and $\sum_N \cap \sum_T = 0$. The set of terminal $\sum_T$ contains the defined {w, l, p, c} of the trajectories assignment.

The non-terminals $\sum_N$ consists of variables, used in words construction of the language body, then the production set P is used for the words construction by operations:

$$\eta \to \gamma \tag{2}$$

According to boundaries, given by means of the productions [8], four grammar types were distinguished: unrestricted, context-sensitive, context-free and regular.

The description of the road traffic manoeuvre the context-sensitive grammar uses productions expressed by the following relations:

$$\eta_1 A \eta_2 \rightarrow \eta_1 \gamma \eta_2 \qquad (3)$$

where: $\eta_1, \eta_2 \in \sum^*, \gamma \in \sum^+, A \in \sum_N$,

The symbol A can be replaced by an empty sequence of symbols γ, when A appears in context of symbols η_1 and η_2 .

4 The traffic description by context-sensitive grammar

The context-sensitive grammar defines the rules of membership, defining an advanced level of words formation of transportation objects free modelling.

It allows describing the manoeuvres as: a vehicle's lane turnover, overtaking and U-turning (back).

The turning manoeuvre, for the grammar construction of the turning radius, is observed in a range of 5 to 25 meters. It concerns identification process of the manoeuvre on crossroads in cities.

The grammar expression for the manoeuvre of the vehicles turning - left or right is expressed as:

$$G_{turn\_left} = \left(\sum_N, \sum_T, P, S\right)_{turn\_left} \qquad (4)$$

where:

$\sum_N$ ={S, A, B, C }, $\sum_T$ = { l, w },

P = :

0. S → All
1. All → lBl
2. Al → Awl
3. Awl → wwlB
4. Awl → wlB
5. lBl → llC
6. Bl → Bwl
7. Bwl → wwlC
8. Bwl → wlC
9. lC → lwC
10. lwC → lwwC
11. C → l

Using the defined rules of membership in P it allows us getting 27 words of the alphabet $\sum$. The largest lock radius contains the lock primitives l of vehicle, separated by two primitives of a direct movement w .

A graphical interpretation of this analysis, concerning the grammar words of the lock G_{turn_left}, has been presented in Fig. 4. The lock symbol l indicated in the figure, concerns a radius $R=5m$. The grammar constructions of the vehicle's turn to the left and to the right, look the same; as in grammar G_{turn_left}.

Figure 6. A grammar G_{turn_left}, for radius R=5m - graphical illustration

The traffic lane change (replacement) is a next important manoeuvre, for the transportation network state description.

The word generation of the manoeuvres' grammar the necessary set of the vehicle's movement symbols has to be assigned.

The description language construction contains units for: driving ahead and the vehicle's lock. The sequence of symbols for the traffic lane change consists of contrary primitives of the vehicle's lock.

For avoiding mistakes in manoeuvres description, in the traffic lanes change, the driving direction descriptors were provided (Fig. 5).

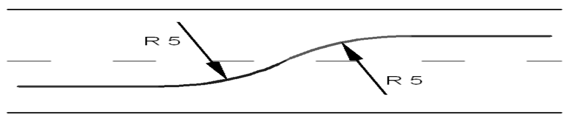

Figure 7. The traffic lanes change manoeuvres – for radius R=5m

The grammar of the traffic lane manoeuvre to the left, define following symbols:

$$G_{change\_left} = \left(\sum_N, \sum_T, P, S\right)_{change\_left} \qquad (5)$$

where:

$\sum_N$ ={S, A}, $\sum_T$ = {l, w, p},

P = :

0. S → wwlApwww
1. Apw → Appw
2. Ap → Awp
3. Awp → Awwp
4. Awwp → Awwwp
5. lA → Al
6. Alw → Allw
7. Alp → Allp
8. Al → wl

The belonging rules P for the traffic lane change allow us finding up to 16 words of the state description, by the alphabet $\sum$.

The manoeuvre shape for the traffic lane change illustrates Fig 6. The dashed area defines a surface of an eligible zone of the traffic lane change: G_{change_left}.

Figure 8. The traffic lane change illustration (G_{change_left})

The grammar constructions for traffic lane change were defined similarly to the above manoeuvre of turning left.

This construction of the manoeuvres' grammar is used for the description at the parking by a kerb, as well. The driving dynamic analysis is indicating a type of the manoeuvre.

Next two vehicle's movement symbols of the language, for overtaking and turning of the vehicle, were considered as well. Their grammar construction was used earlier as elements for the vehicles' manoeuvres assignment. The trajectories of these manoeuvres can also be defined in similar way.

The vehicle's overtaking manoeuvre [1] is defined by the traffic lane change, similarly to driving strait-on with overtaking of a barrier with U-turn (back) and a lane change. The overtaking from the left is defined by:

$$L_{turnover\_left} = \begin{Bmatrix} L_{change\_left}, w^n, L_{change\_right} : \\ 0 \le n \le 4 \end{Bmatrix} \quad (6)$$

where:

L_{change_left} – the traffic change manoeuvre for grammar G_{change_left},

w^n – sequence of primitives; in the form of: { λ, w, ww, www, wwww},

L_{change_right} – the manoeuvre for traffic lane change, defined by a grammar G_{change_right}

The construction $L_{turnover\_left}$ of the language gives us 1280 words, of the manoeuvre of overtaking description (the lane change – forward driving – the lane change).

The right-sided manoeuvre of overtaking was defined analogously to the left-sided manoeuvre with different order of its components:

$$L_{\text{turnover\_right}} = \begin{Bmatrix} L_{change\_right}, w^n, L_{change\_left} : \\ 0 \le n \le 4 \end{Bmatrix} \quad (7)$$

The vehicle's U-turning manoeuvre, in the syntactic description method of the movement uses the early defined language units. The offered solution allows us identify the direct U-turn of the vehicle, often with U-turn manoeuvre.

The language of U-turn manoeuvre description has been defined as follows:

$$L_{U\_turn} = \begin{Bmatrix} L_{turn\_left}, c^n, L_{turn\_left} : \\ 0 \le n \le 4 \end{Bmatrix} \quad (8)$$

where:

L_{turn_left} – the vehicle turn manoeuvre was described by the grammar G_{turn_left}.

c^n – primitives sequence: { λ, c, cc, ccc, cccc},

Number of the language words assigning the turning back manoeuvre contains 3645 sequences with the syntactic symbols of the method (lock – U-turn – lock: $27 \cdot 5 \cdot 27$).

5 The automata for vehicle's movement trajectories description recognition

The automata for the description words accepting [3] the trajectory description language, allows us identify the observed scenery; by a Turing machine and its versions, as: a finite-state automata, push-down automata and linear-bounded automata [7].

This way the language description units are recognised. The machine is counting the function values for the given arguments [7].

The fundamental Turing machine model is assigned by the system [1]:

$$M = \left(Q, \Sigma, \Gamma, \delta, q_0, B, F\right) \quad (9)$$

where:

Q – is finite set of states,

Σ – an input variables set,

$\Sigma \subset \Gamma - \{B\}$ – (the input variables alphabet),

Γ – finite set of eligible bend symbols (the bend alphabet),

δ – the transition function,

q_0 – the starting state, $q_0 \in Q$,

B – blank symbol, $B \in \Gamma$,

F – the accepted states set, $F \subseteq Q$.

5.1 Linearly-bounded automata for states description

The linearly-bounded automata were defined by the Turing machine. It is restricted to modifications of the machine for doing calculations for a part of the bend, where an input status was recorded. The linearly bounded automata is called single-bend Turing machine, with a stop mode. Its alphabet contains two special symbols: starting and ending delimiters that define direct cells with the input data.

The presented work introduces the programming implementation of the automata, constructed as a parser, corresponding with the procedure of the syntactic assignment of traffic accidents in the transportation network.

The linearly-bounded automata define a not deterministic model. The parser constructions for identifying the network states can be expressed by the states:

$$P = (Q, \Sigma, \delta, q_0, F) \tag{10}$$

where:

Q – finite, not empty set of states,

Σ – finite inputs alphabet,

δ – the transition function (for mapping),

q_0 – starting delimiter,

F – ending delimiter.

The defined parsers of the road incidents identifiers [11] concern manoeuvres: turning the vehicle, traffic lane change, overtaking and U-turn of vehicle. The parser construction for a turn left manoeuvres is assigned by the expressions:

$$P_{turn\_left} = (Q, \Sigma, \delta, q_0, F)_{turn\_left} \tag{11}$$

where:

Q – { q0, q1, q2, q3, q4, q5, q6, q7, q8},

Σ – { w, l },

δ – is defined by graph 1 (fig.7),

q_0 – {q0},

F – {q9}.

The linearly-bounded automaton P_{turn_left} allows us finding twenty seven words of the alphabet Σ for vehicles turning manoeuvres.

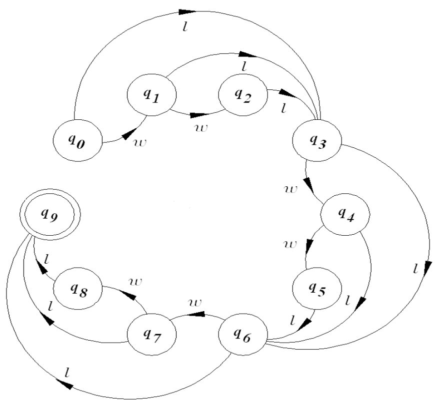

Figure 9. The parser states graph P_{turn_left} for assigning vehicles' turning manoeuvres

The parser states graph P_{turn_left} was presented in Fig. 9.

The manoeuvre of the traffic lanes changing, for a left sided traffic, will be widening the traffic description symbols and a traffic network assignment. Its localisation will identify a parser P_{change_left} , as:

$$P_{change\_left} = (Q, \Sigma, \delta, q_0, F)_{change\_left} \tag{12}$$

where:

Q – {q0, q1, q2, q3, q4, q5, q6, q7, q8, q9, q10, q11, q12, 12, q13, q14},

Σ – {w, p, and l},

δ – is defined by graph 2 (fig.8),

q_0 – {q0},

F – {q14}.

The offered construction P_{change_left} allows us accepting all sixteen alphabet words from Σ ; the language of the manoeuvres, describing the vehicle's movement of the vehicles. The parser P_{change_left} of the traffic states was introduced in Fig 10.

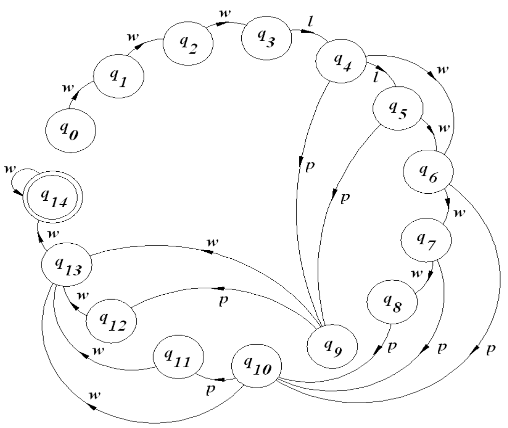

Figure 10. The parser states graph P_{change_left} for vehicles changing manoeuvres

The vehicles overtaking, it is a next manoeuvre that needs to be assigned for a complete transportation network description. The construction of the parser $P_{turnover\_left}$ accepting this trajectory has been defined by the expression:

$$P_{turnover\_left} = (Q, \Sigma, \delta, q_0, F)_{turnover\_left} \tag{13}$$

where:

Q – {q0, q1, q2, q3, q33},

Σ – {w, l, and p},

δ – defined by the graph 3 (in Fig.9),

q_0 – {q0},

F – {q33}.

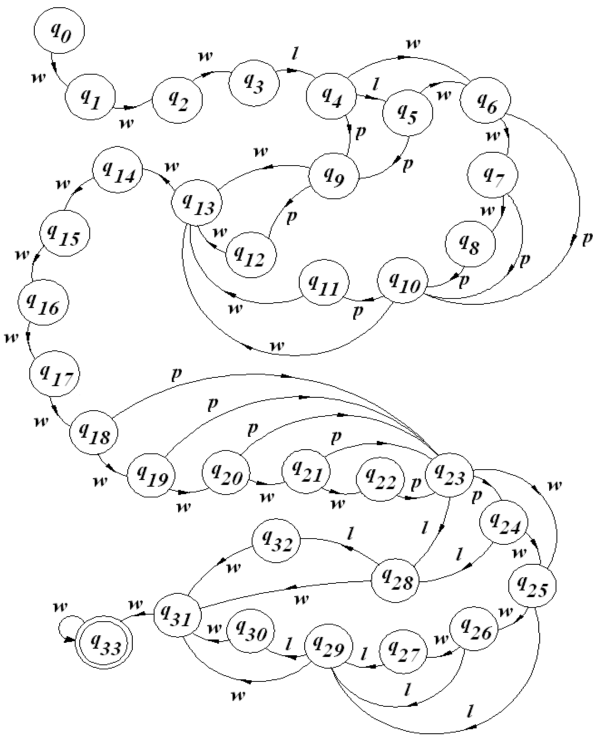

Figure 11. The parser states graph $P_{turnover\_left}$ for the manoeuvres turning-over

A very big number of expressions (1280) defining the complete manoeuvre set for overtaking is a reason that the table of the accepted words by the parser $P_{turnover\_left}$ was omitted.

The next state of the road (with a complex movement trajectory) concerns the U-turn of the vehicle.

The parser P_{U-turn} proposal uses early defined automata for the turn-right and turn-left manoeuvres, like in equation bellow:

$$P_{U-turn} = \left(Q, \Sigma, \delta, q_0, F\right)_{U-turn} \qquad (14)$$

where:

Q – {q0, q1, q2, q3, …, q23},

$\sum$ – {w, p, c},

δ – is defined by the graph 4 (in Fig.10),

q_0 – {q0},

F – {q23}.

Remarkable big number of words (3645), defining the U-turn manoeuvre, makes it impossible to illustrate this operation by a table of word accepted by the parser P_{U-turn}.

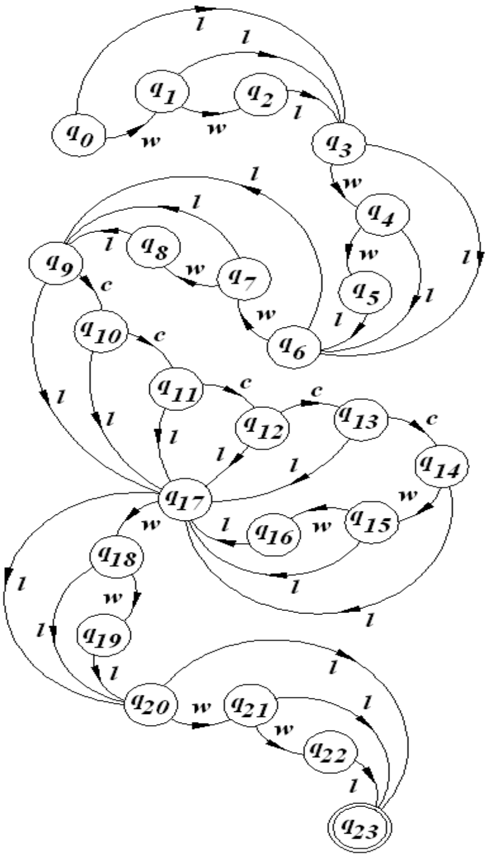

Figure 12. The parser states graph P_{U-turn} for vehicles U-turn manoeuvres

6 Conclusions

The vehicles’ movement trajectories description allows us converting, very complex formalisms for states set of the road traffic. They convert a very difficult transportation scenery layout into the sequence of video frames, with simplified syntactic description of this scene.

The trajectories symbols define all movements of vehicles at the video sequences.

Today’s attention given the video-recorders comes from very useful data available for various analyses (fashionable applications). They provide the traffic control technologies with unique data sets supporting the traditional data units; as number of passing vehicles with their characteristics, as: speed, vehicles classes, specific traffic services, etc. It is remarkable more then it is available by classical measurements, with inductive loops or photo radars.

The elaborated, by the contribution’s authors, algorithms allow us not only detect the passing vehicles. The method identifies the vehicles movement trajectories with many traffic accidents, visible in an objective of the camera.

This way very advanced traffic control algorithms can be defined and executed.

The elaborated methods of the video-image registration were verified empirically in time of the project, for the video technologies transfer into small industry, carried on within the EC funds actions (in years 2006-2008).

The data for analysis is recorded by video real time systems, installed on several crossroads of the city of Katowice. They are powered by traffic controllers of ZIR-SSR Bytom, a leading producer of the road traffic controllers in Poland.

All step of the development works were verified continuously in time of the project development stages, by a research team in Informatics Systems Department of Transport, at Faculty of Transport.

The introduced methods can be extended into further incidents of the road traffic description. What is more the available data set allows us to use modern video technologies for traffic identification, description, control and management; recorded by a single video-camera unit or by remarkable limited number of the cameras, watching the traffic scene of big cities.

7 References

[1] Aho A., V., Ullman J., D.: **The theory of parsing, Translation and Compiling,** Prentice-Hall Series in Automatic Computation, Vol. 1, 1972.

[2] Beymer D., McLauchlan P., Coifman B., Malik J.: **A Real-time Computer Vision System for Measuring Traffic Parameters,** Proc. of Int. Conf. Computer Vision and Pattern Recognition Workshops , p. 495-501, 1997

[3] Chomsky A.: **Syntactic structure,** Published: Walter de Gruyter, Berlin, 2002.

[4] Deng L., Tang N., Lee D., Wang Ch., Lu M.: **Vision Based Adaptive Traffic Signal Control System Development,** Proc. Of Int. Conf. Parallel and Distributed Systems, Vol.2, p 634 – 638, 2005

[5] Fu K. S.: **Syntactic Methods in Pattern Recognition,** Academic Press, New York, 1974.

[6] Fu K. S.: **Syntactic Pattern Recognition and Applications**, New Jersey, Prentice-Hall, 1982.

[7] Hopcroft J., Motwain R., Ullman J.: **Introduction to Automata Theory, Languages and Computation,** Person Education, Second Edition, 2001.

[8] Kotowski W.: **Kodeks drogowy - wprowadzenie,** (in polish), Stan prawny 2002-01-01, Wydawnictwo Infor, Warszawa 2002.

[9] Piecha J., Staniek M.: **The syntactic alphabet analysis for vehicles simple and fast assignment,** Proc of Int. Conf. on Telematics, Logistics & Transport Safety, Silesian University of Technology, Katowice, 2009.

[10] Roess R., McShane W., Prassas E.: **Traffic Engineering,** New York, Prentice-Hall, 2004.

[11] Staniek M.: **Metoda syntaktycznego opisu trajektorii ruchu pojazdów w sieci transportowej**, PhD work (in polish), Gliwice 2009.