

Jarosław Wąs*, Rafał Bieliński*, Bartłomiej Gajewski*, Patryk Orzechowski*

Problematyka modelowania ruchu miejskiego z wykorzystaniem automatów komórkowych

1. Wprowadzenie

W ostatnich latach możemy zaobserwować znaczący wzrost zainteresowania modelowaniem ruchu miejskiego. Dotyczy to zarówno ruchu pieszych, jak i ruchu samochodowego. Należy zauważyć, że przeprowadzenie skutecznych i realistycznych symulacji komputerowych pozwala na sprawdzenie konkretnych rozwiązań technicznych już na etapie projektu. Podobnie rzecz ma się z przeprowadzanymi remontami dróg oraz organizowaniem objazdów, dużo prościej jest zaplanować wszelkie prace remontowe, korzystając z symulacji komputerowej, niż testować to na żywej tkance organizmu miejskiego.

1.2. Sformułowanie problemu

Celem niniejszej pracy jest zaproponowanie modelu samochodowego ruchu miejskiego stworzonego z wykorzystaniem istniejących modeli. Model ruchu powinien uwzględniać podejście mikroskopowe w modelowaniu, tzn. uwzględniać specyfikę ruchu pojedynczego pojazdu poruszającego się po drogach jednopasmowych oraz wielopasmowych. Model ma uwzględniać specyfikę zmiany pasa ruchu, brać pod uwagę charakterystyki przyspieszania oraz hamowania pojazdu. Ponadto powinien mieć możliwości użycia mapy drogowej miasta w formacie cyfrowym do szczegółowego zamodelowania topografii miasta.

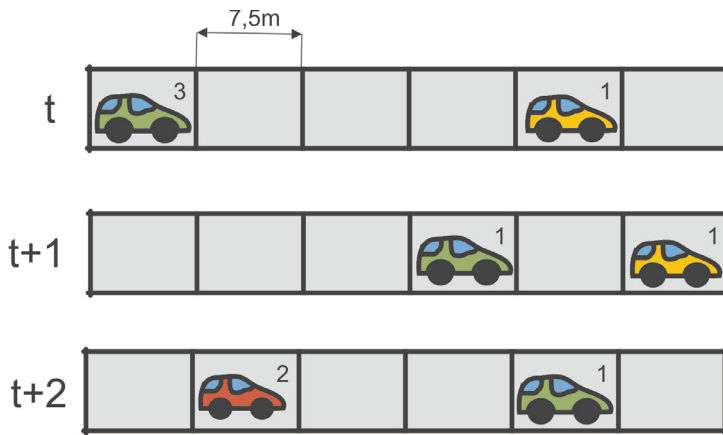
1.3. Klasyczny model mikroskopowy ruchu drogowego

Automaty komórkowe CA można przedstawić w postaci czwórki (L, S, N, f) . Gdzie poszczególne elementy oznaczają: L – przestrzeń podzieloną na siatkę cel, S – zbiór skończonych stanów, N – zbiór sąsiadów danej celi, f – funkcja zmiany konfiguracji w poszczególnych celach. Konfiguracja $C_t: L \rightarrow S$ jest funkcją, którą łączy stan z każdą celą siatki. Funkcja f zmienia konfigurację C_t w nową konfigurację C_{t+1} . Dokładniejszy opis modelowania za pomocą automatów komórkowych można znaleźć przykładowo w pracach [10] czy [17].

* Katedra Automatyki, Akademia Górniczo-Hutnicza w Krakowie

Klasycznym modelem opartym na automatach komórkowych, opisującym ruch samochodowy, jest model Nagela–Schreckenberga [8] (nazywany również krótko *Na-Sch*). Model ten dotyczy ruchu samochodów na autostradzie, lecz jest punktem odniesienia również dla symulacji ruchu miejskiego. W modelu *Na-Sch* przyjęto rozmiar komórki równy $d = 7,5$ m. Prędkość pojazdu opisywana jest liczbą komórek pokonywanych przez pojazd w chwili czasowej (rys. 1). Model opisują następujące reguły ruchu:

- Przyspieszenie: $v(t+1) \rightarrow \min(v(t) + 1, v_{\max})$, gdzie $v(t)$ to prędkość aktualna.
- Hamowanie: $v(t+1) \rightarrow \min(v(t), g(t) - 1)$, gdzie $g(t)$ jest liczbą pustych komórek pomiędzy pojazdami.
- Element losowy (losowe hamowanie): prawdopodobieństwo p , że zajdzie $v(t+1) \rightarrow \max(v(t) - 1)$, jeżeli $v(t) \geq 1$.
- Ruch (zmiana położenia w czasie): $x(t+1) = x(t) + v(t)$.



Rys. 1. Ruch w modelu Nagela–Schreckenberga na pasie ruchu w kolejnych chwilach czasowych. W lewym górnym rogu każdej komórki możemy zaobserwować aktualną prędkość pojazdu wyrażoną w liczbie komórek pokonywanych w jednej chwili czasowej

Na bazie modelu Nagela–Schreckenberga – stworzono szereg modeli odnoszących się do ruchu miejskiego. Wśród tych prac należy wymienić pracę Essena i Schreckenberga [3], w której autorzy zaproponowali m.in. reguły zmian pasa ruchu. Z kolei Dupuis i Chopard zaproponowali równoległą implementację ruchu drogowego na przykładzie organizacji ruchu w Genewie [2]. Wśród prac bazujących na modelu *Na-Sch* należy wymienić pracę Hartmana opisującą między innymi ruch z uwzględnieniem rond oraz skrzyżowań [4]. Z kolei praca Płaczka [16] podejmuje bardzo ważny problem praktycznej akwizycji zmiennych do modeli opartych na modelach komórkowych, z kolei zagadnienia związane z kontrolą ruchu opisuje np. praca [13].

Ciekawą pracą nawiązującą do praktycznych badań jest praca [11] opisująca szereg danych rzeczywistych dotyczących modelowania ruchu. Podobnie nowatorskie podejście w modelowaniu ruchu oparte na systemach rozmytych opisują prace [13] oraz [16].

W dalszej części pracy autorzy szczegółowo nawiązują do poszczególnych modeli opisanych w literaturze, w szczególności opierając się na metodach opartych na automatach komórkowych.

2. Zmodyfikowany model Nagela–Schreckenberga dla ruchu w mieście

Jedną z głównych wad modelu *Na-Sch* jest fakt, iż dynamika pojedynczego samochodu nie jest dobrze odwzorowywana [9]. Przyśpieszenie pojazdu może przyjmować zbyt duże wartości, które odbiegają znacznie od rzeczywistości, a drugi krok modelu opisujący hamowanie służy tylko zapobieganiu kolizjom i zapewnieniu odpowiedniego odstępu między samochodami, nie odwzorowując przy tym rzeczywistego zachowania pojazdu podczas zwalniania. W konsekwencji wartości związane z ruchem samochodu (jak zużycie paliwa, emisja spalin) nie są dobrze opisywane [9]. Należy również zauważyć, że model *Na-Sch* powstał w celu symulacji przepływu pojazdów na autostradzie. Ruch miejski ma trochę inny charakter – polega na ciągłym hamowaniu i przyśpieszaniu, podczas gdy pojazdy na autostradzie utrzymują przez większą część czasu stałą prędkość.

Proponowany model kładzie większy nacisk na kwestię dokładniejszego opisu ruchu pojedynczego samochodu. Wprowadzono w nim zmienne przyśpieszenie pojazdu, które w szczególności może przyjmować wartości mniejsze niż w modelu *Na-Sch* oraz stopniowe zwalnianie przed przeszkodą i poprzedzającym samochodem.

W stosunku do modelu *Na-Sch* proponujemy inny sposób reprezentacji pojazdu na siatce automatu. W oryginalnym modelu samochód zajmuje zawsze pojedynczą komórkę. Rozwiązanie autorów zakłada natomiast, że pojedynczy pojazd może zająć większą ich liczbę. Dzięki temu pojedyncza komórka nie musi już odpowiadać odcinkowi rzeczywistej drogi o długości 7,5 m. Większa elastyczność w doborze skali modelu pozwala między innymi na osiągnięcie lepszego odwzorowania rzeczywistej prędkości samochodu niż w modelu *Na-Sch*. Proponowany model uwzględnia również wielopasmowe drogi oraz skrzyżowania ze światłami.

2.1. Model w mikroskali

Podobnie jak w pracy D. Hartmana [4], jednokierunkową, wielopasmową drogę, będącą podstawową strukturą sieci drogowej, reprezentuje siatka kwadratowych komórek. Każdy „rząd” komórek odpowiada pojedynczemu pasowi drogi. Sposób reprezentacji pojazdu i jego poruszania się po prezentowanej siatce jest zgodny z algorytmem *Leading Head Algorithm* (LHA) [9]. Pojazd jest reprezentowany za pomocą zajętości kilku sąsiednich komórek, które w dalszej części pracy będziemy nazywać skrótowo członami bądź segmentami pojazdu. Ruch rozpoczyna pierwszy segment pojazdu, poruszając się o jedną komórkę w daną stronę. Każdy następny człon samochodu zajmuje miejsce poprzedniego (rys. 2).

czeństwa samochodu wykonują manewr. Wartość funkcji jest uzależniona od warunków na drodze opisywanych przez następujące wartości:

- prędkość samochodu zmieniającego pas V_i ,
- prędkość najbliższego pojazdu z tyłu na pasie docelowym V_p ,
- odległość między samochodami S_t .

Wartości te pokazane są na rysunku 2 (rozpatrywany jest niebieski pojazd). Przykładowa funkcja p_c znajduje się poniżej:

$$p_c(V_t, S_t) = \begin{cases} 0,2, & \text{gdy } V_t < S_t \\ 0,9, & \text{gdy } V_t \geq S_t \end{cases}.$$

W przypadku kiedy prędkość pojazdu znajdującego się z tyłu na pasie, na który zamierza wjechać rozpatrywany samochód, jest większa niż liczba wolnych komórek ($V_t < S_t$), pojazd z tyłu będzie zmuszony do hamowania w aktualnej turze. Definiując funkcję, uznano taką sytuację za niepożądaną, dlatego zmiana pasa w takich warunkach wykonywana jest z niskim prawdopodobieństwem wynoszącym 0,2. Jeżeli takie okoliczności nie mają miejsca, pojazd zmienia pas z prawdopodobieństwem 0,9. Prędkość samochodu zmieniającego pas nie jest w brana pod uwagę, jednak można sobie wyobrazić funkcję, która uwzględniłaby to, jak szybko samochody zbliżają się do siebie, a więc również wielkość V_i .

Podobnie jak w modelu *Na-Sch*, pojazd w każdej iteracji porusza się zgodnie z kierunkiem jazdy o całkowitą liczbę komórek, jednak aby dokładniej opisać jego dynamikę, prędkość i przyspieszenie wyrażone są przez liczby zmiennoprzecinkowe, a nie całkowite, jak ma to miejsce w oryginalnym modelu. W każdym kolejnym kroku, prędkość jest zwiększana o wartość przyspieszenia, a całkowitą liczbę komórek, o jaką przesunie się pojazd w danej iteracji, określa wartość prędkości po odrzuceniu jej części ułamkowej.

W rzeczywistości przyrost prędkości samochodu następuje bardziej gwałtownie przy mniejszych prędkościach i spada w miarę, jak pojazd się rozpędza. Jako że jednym z celów modelu było umożliwienie jak najlepsze odwzorowanie zachowania się pojedynczego samochodu, postanowiono wprowadzić funkcję $a(v)$ opisującą przyspieszenie w funkcji prędkości, którą można dobrać tak, aby opisywała ona sposób przyspieszania rzeczywistych pojazdów.

Uwzględniając maksymalną prędkość i -tego samochodu w chwili czasowej t , jego prędkość w chwili $t + 1$, obliczaną w czwartym kroku modelu, można przedstawić w następujący sposób:

$$V(i, t+1) = \min \left(\{V_{\max}(i, t), V(i, t) + a(v(i, t))\} \right).$$

Prędkość maksymalna $V_{\max}(i, t)$ uwzględnia ograniczenia prędkości na danym odcinku (znaki oraz ograniczenia wynikające z rodzaju drogi), pozycję samochodu poprzedzają-

cego (kolizje są niedozwolone) oraz sposób, w jaki samochód zwalnia w następujących sytuacjach:

- przed nieruchomą przeszkodą (końcem pasa, czerwonym światłem, nieruchomym pojazdem),
- przed wolniej poruszającym się samochodem,
- ograniczeniem prędkości.

Charakterystykę zwalniania samochodu najprościej można opisać, przedstawiając jego maksymalną prędkość $f_z(s)$ w funkcji odległości od nieruchomej przeszkody. Odpowiednio dobierając tę funkcję, możemy kontrolować zachowanie się pojazdów podczas zwalniania. Resztę podanych wyżej przypadków można w prosty sposób sprowadzić do sytuacji opisywanej przez funkcję f_z .

Dla zwalniania jednostajnego przed przeszkodą z opóźnieniem a_b funkcję f_z można przedstawić następująco:

$$f_z(s) = \begin{cases} 0, & \text{dla } s = 0 \\ \max\left(\left\{1, \min\left(\left\{s, \sqrt{2sa_b}\right\}\right)\right\}\right) & \text{dla } s > 0 \end{cases}$$

Wartość $\sqrt{2sa_b}$ określa prędkość w ruchu jednostajnie opóźnionym w zależności od odległości. Powyższa funkcja uwzględnia dodatkowo następujące ograniczenia: pojazd nie może przejechać przez przeszkodę ($f_z(s) \leq s$) oraz mając przed nią przynajmniej jedną komórkę wolnego miejsca, musi wykonać ruch ($f_z(s) \geq 1$, gdy $s \geq 1$).

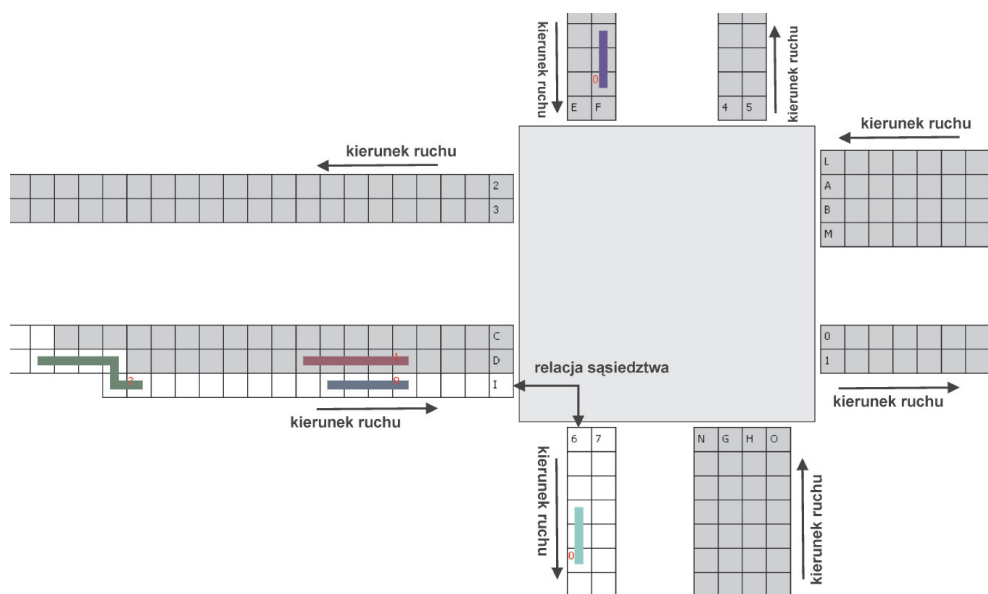
Krok uwzględniający losowość jest analogiczny do tego w modelu *Na-Sch* – prędkość samochodu jest obniżana o 1 z prawdopodobieństwem p_z .

W ostatnim kroku następuje przesunięcie pojazdu zgodnie z kierunkiem ruchu, o tyle komórek, ile wynosi wartość prędkości samochodu po odrzuceniu jej części ułamkowej.

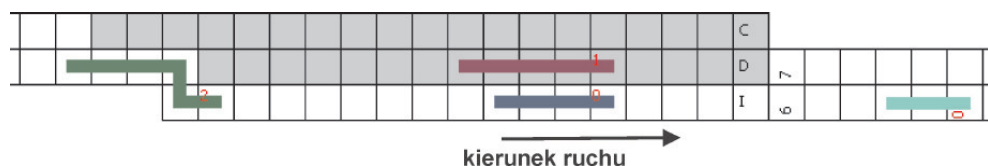
2.2. Sieć drogowa i skrzyżowania ze światłami

Sieć dróg może zostać przedstawiona w formie grafu skierowanego, w którym węzły reprezentują skrzyżowania, a krawędzie jednokierunkowe odcinki drogi pomiędzy nimi. Specjalnym rodzajem wierzchołków są generatory tworzące samochody na granicy modelowanej sieci drogowej i terminatory usuwające pojazdy wyjeżdżające poza modelowaną przestrzeń. Graf może być stworzony na podstawie mapy cyfrowej GPS (co zostało opisane w rozdz. 3) i na tej podstawie zbudowana jest siatka automatu komórkowego. Zgodnie z rozwiązaniem opisanym w pracy Essera i Schreckenberga [1] ruch pojazdów na samym skrzyżowaniu nie jest modelowany, a jedynie definiowane są wejścia i wyjścia. Pojazdy przejeżdżają bezpośrednio z jednego odcinka drogi na drugi. Ruch ten można kontrolować poprzez zmianę sąsiedztwa skrajnych komórek należących do danych odcinków dróg. Sytuacja, w której komórki sąsiadują ze sobą, jest równoznaczna z wystąpieniem zielone-

go światła. W momencie, kiedy dana komórka nie ma zdefiniowanego sąsiedztwa w kierunku ruchu, stanowi koniec pasa (czerwone światło) – pojazd zatrzyma się na niej zgodnie z regułami kontrolującymi prędkość. Zatem skrzyżowanie jest reprezentowane w tym przypadku w postaci zbioru reguł, które określają, jak zmieniać w czasie sąsiedztwo odpowiednich komórek (rys. 3, 4). Ustąpienie pierwszeństwa przejazdu może zostać zrealizowane poprzez dodatkową zmianę światła (sąsiedztwa) w zależności od warunków na drodze.



Rys. 3. Przykładowe skrzyżowanie



Rys. 4. Skrzyżowanie z rysunku 3 z punktu widzenia zielonego pojazdu.
Szare komórki nie są dla niego dostępne

3. Wykorzystanie map cyfrowych w tworzeniu modeli

W ostatnich latach duże znaczenie uzyskały mapy w formacie cyfrowym. W modelach miejskich, które swym zasięgiem muszą obejmować nieraz gęstą sieć dróg, ważne jest uzyskanie jak największej ilości informacji z dostępnych map cyfrowych. W tym celu auto-

rzy skupili swoją uwagę na znajdującym coraz większe uznanie formacie Polish Map Format i zastosowali go w modelu symulacji ruchu drogowego.

3.1. Polish Map Format

Polski Format Mapy (*Polish Format*) to jeden ze sposobów reprezentacji danych geograficznych w postaci kodu źródłowego. Zakodowane w ten sposób informacje dotyczące typu oraz położenia geograficznego poszczególnych obiektów można swobodnie konwertować przy użyciu specjalnych programów z pliku tekstowego (o standardowym rozszerzeniu .MP) do różnorodnych formatów – m.in. *Keyhole Markup Language* (.KML) używanego przez Google Earth i Google Maps [7], czy też do zamkniętego formatu portowalnego na urządzenia GPS firmy Garmin (.IMG) [1].

Nazwa *Polish Format* historycznie pochodzi od narodowości Stanisława Kozickiego – twórcy programu cGPSmapper, pozwalającego na konwersję map zapisanych w formie tekstowej na urządzenia GPS.

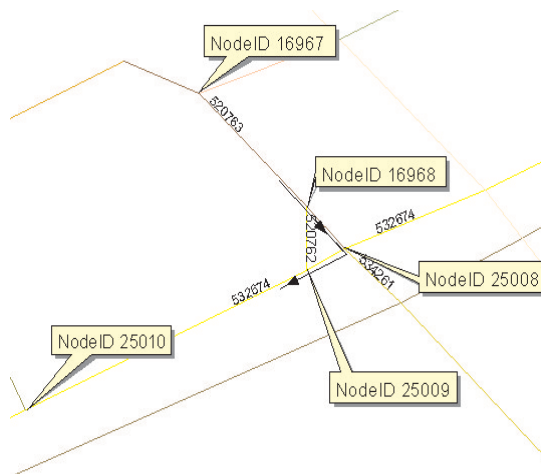
Specyfikacja techniczna formatu wymusza określoną postać początkowych linii pliku .MP jak również opisu poszczególnych obiektów, które reprezentuje jeden z kilku podstawowych typów, m.in. POLYGON (m.in. jeziora, budynki), POLYLINE (m.in. ulice, drogi kolejowe), RESTRICTION (nakazy/zakazy skrętu), czy POI (*Point of Interest* – ważniejsze obiekty, np. teatry, szpitale) przykładowo:

```
[POLYLINE]
RoadID=100
Type=0x5
Label=FLORIANSKA
EndLevel=2
RouteParam=3,1,0,0,1,1,1,1,1,0,1,1
Dta0=(50.064830,19.941300),(50.063640,19.940420),
(50.062900,19.939800),(50.062060,19.939180)
Nod1=3,4284,0
Nod2=2,4333,0
Nod3=1,4374,0
Nod4=0,4442,0
[END]
```

Poszczególne linie kodu określają parametry techniczne drogi – unikalny identyfikator drogi na mapie (*RoadID*), typ drogi (*Type* – np. droga główna, osiedlowa), nazwę (*Label*), informację o poziomie szczegółowości prezentacji (*EndLevel*). Współrzędne punktów geograficznych tworzących drogę w odpowiedniej skali koduje się w linii zawierającej *DataX*. Współrzędne punktów reprezentowane są z dokładnością do 6 miejsc po przecinku, co pozwala na konwersję z maksymalną precyzją do formatu Garmina równą $360^{\circ}/2^{24} = 2,1 \cdot 10^{-5}$ (ok. 2,4 m) [5].

Wśród punktów określających położenie geograficzne drogi (np. modelujących zakręty) w tagu *Data*, wyróżnia się tzw. punkty węzłowe (*Nody*). Węzły definiują istotne miejsca na drogach – początki i końce ulic, skrzyżowania. *Nody* wskazuje się podając ich pozycję węzłów na liście *Data*, oraz nadając unikalny w skali mapy identyfikator (rys. 5). W formacie możliwe jest zakodowanie dodatkowych parametrów drogi, takich jak dozwolona prędkość, kategoria oraz jedno- lub dwukierunkowość drogi, określone zakazy ruchu (np. dla ciężarówek, rowerów itp.), jak również nakazy ruchu. Więcej szczegółów o formacie można odnaleźć w specyfikacji [1].

Powstanie aplikacji konwertującej mapę spowodowało rozwój programów do edycji map – np. darmowego GPSMapEdit [5] czy komercyjnego GlobalMapper [6]. Pozwoliło także na powstanie amatorskich map wielu miast i rejonów w Polsce i na świecie (m.in. Krakowa, Poznania, Nowego Jorku, Rio de Janeiro), udostępnianych darmowo na stronie [1].

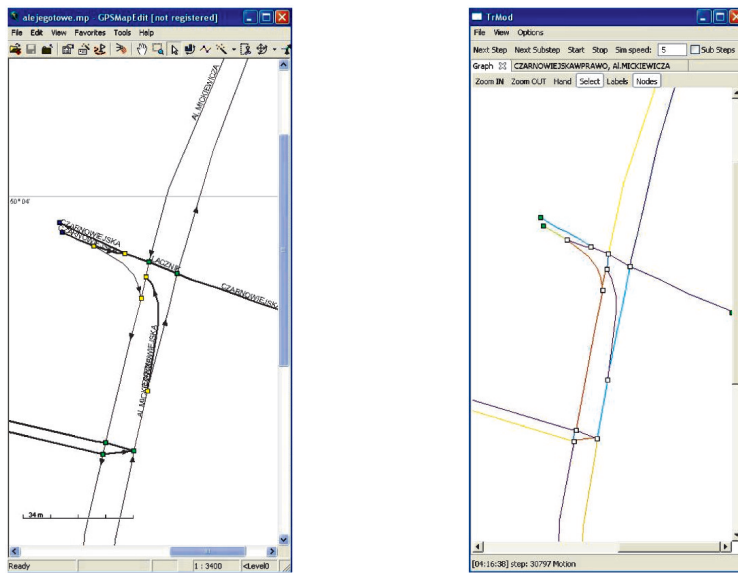


Rys. 5. Fragment mapy z *Polish Map Format*

Odcinek drogi między skrzyżowaniami jest reprezentowany przez prostokątną siatkę komórek (rys. 6). Dla uproszczenia założono, że taka siatka nie uwzględnia rzeczywistej geometrii ulicy, a tylko jej długość. Pojedyncza komórka nie ma więc dokładnie zdefiniowanego położenia na mapie. Można je w przybliżeniu podać na podstawie odległości od skrzyżowania, jednak nie opracowano metody dokładnego naniesienia siatki komórek na mapę gps.

Mapa w formacie *Polish Map* (rys. 6), konwertowana na siatkę komórek w aplikacji TrMod, służy więc głównie do określenia, w których miejscach występują skrzyżowania, jakie drogi do nich dochodzą i jaka jest długość poszczególnych odcinków jezdni (żeby na tej podstawie określić długość wycinka siatki wyrażoną w komórkach). Dodatkowo można wykorzystać informację o ograniczeniach prędkości i zakazach skrętu. *Polish Map Format*

nie zawiera niestety informacji na temat pasów ruchu, dlatego takie dane jak ich liczba, przeznaczenie czy też położenie na jezdni muszą zostać uzupełnione ręcznie w aplikacji *TrMod*.



Rys. 6. Mapa w formacie *Polish Map* przedstawiona w oknie edytora map GPS po lewej oraz w oknie aplikacji *TrMod* implementującej prezentowany model po prawej

4. Podsumowanie

Celem pracy było stworzenie modelu ruchu miejskiego z wykorzystaniem automatów komórkowych, w odniesieniu do modelu Nagela–Schreckenberga oraz innych znanych modeli ruchu miejskiego. Przyjęty sposób reprezentacji przestrzeni, reprezentacji samych poruszających się pojazdów, a także zaproponowane reguły poruszania się tychże pojazdów, umożliwiają prowadzenie realistycznych symulacji. W pracy magisterskiej [12] zamieszczono szereg badań rzeczywistych, opartych na analizie wideo kilku skrzyżowań, które pozwoliły wysnuć pierwsze, optymistyczne wnioski na temat przyjętego modelu.

Przy tworzeniu reprezentacji przestrzeni wykorzystali dostępne mapy cyfrowe w formacie *Polish Map*. Podejście takie pozwala na uzyskanie wielu danych w sposób automatyczny. Przebieg tworzenia reprezentacji przestrzeni w modelu ma zatem dwuetapowy charakter: akwizycja danych z *Polish Map Format* oraz doprecyzowanie szczegółów i dostosowanie do wymogów modelu.

Zaproponowany w pracy model wymaga oczywiście dalszych testów i przeprowadzenia pełnej walidacji.

Literatura

- [1] CGPS Map <http://cgpsmapper.com>.
- [2] Dupuis A., Chopard B., *Parallel simulation of traffic in Geneva using cellular automata*. Virtual Shared Memory For Distributed Architectures, Nova Science Publishers, Commack, NewYork, 2001, 89–107.
- [3] Esser J., Schreckenberg M., *Microscopic Simulation of Urban Traffic Based on Cellular Automata*. International Journal of Modern Physics, vol. 8, No. 5, 1997, 1025–1036.
- [4] Hartman D., *Head Leading Algorithm for Urban Traffic Modeling*. Proceedings 16th European Simulation Symposium, 2004.
- [5] Geopainting <http://www.geopainting.com>.
- [6] Globalmap <http://cgpsmapper.com>.
- [7] Gpslib <http://www.gpslib.net/services/Polish-to-Google.Earth/>.
- [8] Nagel K., Schreckenberg M., *A cellular automaton model for freeway traffic*. J. Phys. I France 2, 1992, 2221–2229.
- [9] Wagner P., *Traffic Simulations Using Cellular Automata: Comparision With Reality*. Traffic and Granular Flow 1995, Conference Proceedings.
- [10] Wąs J., Gudowski B., *Zastosowanie automatów komórkowych w symulacji dynamiki pieszych*. Automatyka (półrocznik AGH), t. 8, z. 3, 2004, 303–314.
- [11] Treiber M., Hennecke A., Helbing D., *Congested Traffic States in Empirical Observations and Microscopic Simulations*. 2000 <http://arxiv.org/abs/cond-mat/0002177v2>.
- [12] Bieliński R., Gajewski B., *Modelowanie ruchu drogowego z zastosowaniem automatów komórkowych za pomocą zmodyfikowanego modelu Nagela-Schreckenberga*. Katedra Automatyki AGH, 2009 (praca magisterska).
- [13] Adamski A., Habdank-Wojewódzki S., *Fuzzy stochastic model of traffic events*. ITS ILS'07 dokument elektroniczny, 2007.
- [14] Adamski A., Ingram M., *The advanced traffic inspection system*. [w:] Transactions on Transport Systems Telematics : Modelling, Management and Image Processing. Wydawnictwo Politechniki Śląskiej, Gliwice, 2007.
- [15] Płaczek B., *The method of data entering into cellular traffic model for on-line simulation*. [w:] Transactions on Transport Systems Telematics. J. Piecha (red.), Wydawnictwo Politechniki Śląskiej, Gliwice, 2006.
- [16] Płaczek B., *Fuzzy cellular model for on-line traffic simulation*. Konferencja PPAM, Workshop on Complex Collective Systems, Wrocław LNCS Springer-Verlag, 2009 (w druku).
- [17] Kułakowski K., Wąs J., *Two Concurrent Algorithms of Discrete Potential Field Construction*. Konferencja PPAM, Workshop on Complex Collective Systems, Wrocław LNCS Springer-Verlag, 2009 (w druku).