

Załącznik 3

Autoreferat

1. **Imię i nazwisko:** Adam Sędziwy

2. **Posiadane dyplomy, stopnie naukowe:**

- a. **Doktor nauk technicznych** w dyscyplinie informatyka, Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie, Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki, rozprawa doktorska pt. *Automatyczne wykrywanie i analiza zmian patologicznych w obrazach MR i CT struktur mózgowych*, 2003.
- b. **Magister informatyki** (*cum laude*), dziedzina: informatyka – syntaktyczne metody rozpoznawania obrazów, Instytut Informatyki Uniwersytetu Jagiellońskiego, 2000.
- c. **Magister fizyki** (*cum laude*), dziedzina: fizyka teoretyczna – fizyka cząstek elementarnych, Instytut Fizyki Uniwersytetu Jagiellońskiego, 1996.

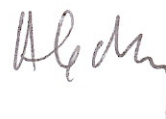
3. **Informacje o dotychczasowym zatrudnieniu w jednostkach naukowych:**

- a. Adiunkt, Katedra Informatyki Stosowanej, Wydział Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej, Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie, 2012–aktualnie.
- b. Adiunkt, Bielska Wyższa Szkoła im. J. Tyszkiewicza, 2007–2012.
- c. Adiunkt, Katedra Automatyki, Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki, Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie, 2004–2012.
- d. Asystent, Katedra Automatyki, Wydział Elektrotechniki, Automatyki, Informatyki i Elektroniki, Akademia Górniczo-Hutnicza im. Stanisława Staszica w Krakowie, 2002–2004.
- e. Asystent, Instytut Informatyki, Wydział Matematyki, Fizyki i Informatyki, Uniwersytet Jagielloński, 2000–2001.

4. **Wskazanie osiągnięcia**

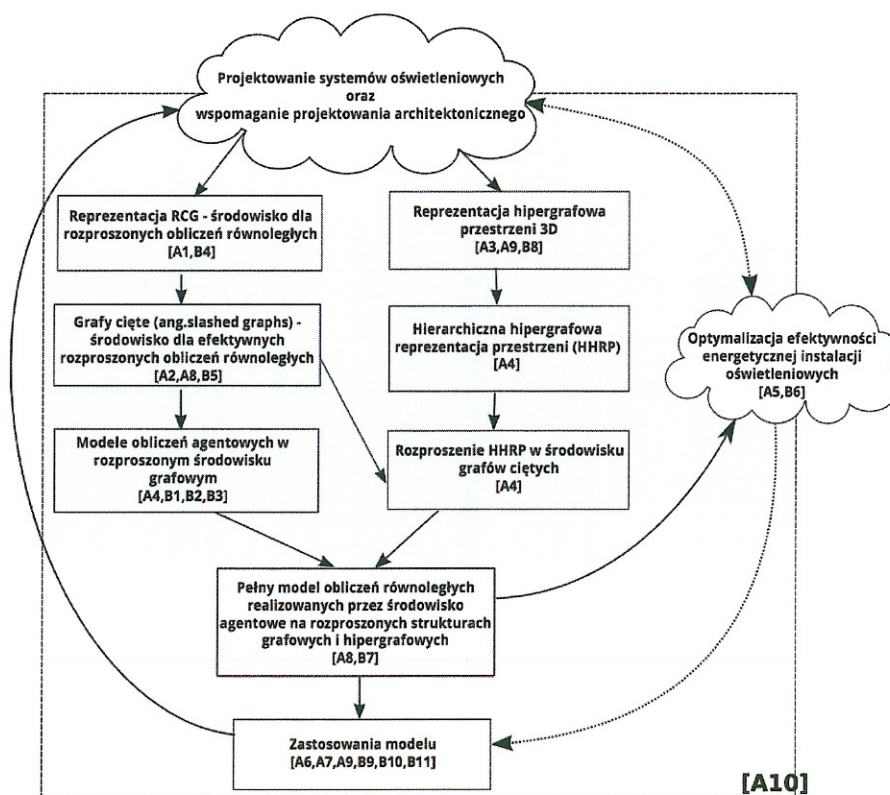
(a) **Tytuł osiągnięcia naukowego:**

Cykl 10 publikacji (w tym monografia podsumowująca wyniki): ***Zastosowanie rozproszonych modeli i transformacji grafowych opartych o reprezentację grafów ciętych (slashed graphs) w problemach projektowania i sterowania***



(b) Autor/autorzy, tytuły publikacji, rok wydania, nazwa wydawnictwa:

- [A1] L. Kotulski, A. Sędziwy, *GRADIS – the multiagent environment supported by graph transformations*, Simulation Modelling Practice and Theory : international journal of the Federation of European Simulation Societies, 2010, Elsevier, DOI: 10.1016/j.simpat.2010.05.013 (IF: 0.736, 13 pkt.)
- [A2] A. Sędziwy, *Effective Graph Representation Supporting Multi-agent Distributed Computing*, International Journal of Innovative Computing, Information and Control, Vol.10, No. 1, February 2014, ICIC International
- [A3] A. Sędziwy, *Representation of Objects in Agent-Based Lighting Design Problem*, Advances in Intelligent and Soft Computing, 2012, Springer, DOI: 10.1007/978-3-642-30662-4_14 (Indeksowane przez Web Of Science, 10 pkt.)
- [A4] A. Sędziwy, L. Kotulski, *Multi-agent system supporting automated large-scale photometric computations*, Entropy 18(3), 76, 2016, MDPI, DOI: 10.3390/e18030076 (IF: 1.502, 30 pkt.)
- [A5] A.Sędziwy, *A New Approach To Street Lighting Design*, LEUKOS: The Journal of the Illuminating Engineering Society of North America, 12(3), 2016, Taylor and Francis, DOI:10.1080/15502724.2015.1080122 (IF: 0.958, 25 pkt.)
- [A6] I. Wojnicki, S. Ernst, L. Kotulski, A. Sędziwy, *Advanced Street Lighting Control*, Expert Systems with Applications, 2014, Elsevier, DOI: 10.1016/j.eswa.2013.07.044 (IF: 2.240, 35 pkt.)
- [A7] A. Sędziwy, L. Kotulski, *Towards highly energy-efficient roadway lighting*, Energies, 9(4), 263, 2016, MDPI, DOI: 10.3390/en9040263 (IF: 2.072, 25 pkt.)
- [A8] A. Sędziwy, *Sustainable Street Lighting Design Supported By Hypergraph-Based Computational Model*, Sustainability , 8(1), 13, 2016, MDPI, DOI:10.3390/su8010013 (IF: 0.942, 15 pkt.)
- [A9] L. Kotulski, A. Sędziwy, B. Strug, *Heterogeneous Graph Grammars Synchronization in Computer Aided Design*, Expert Systems with Applications, 2014, Elsevier, DOI: 10.1016/j.eswa.2013.07.043 (IF: 2.240, 35 pkt.)
- [A10] A.Sędziwy, *Grafowe środowisko obliczeniowe dla obliczeń równoległych i jego zastosowanie w problemie projektowania oświetlenia*, ISBN 978-83-7464-854-7, 2016, Wydawnictwa AGH



Rysunek 1: Przebieg badań nad pełnym modelem formalnym i jego zastosowaniami.

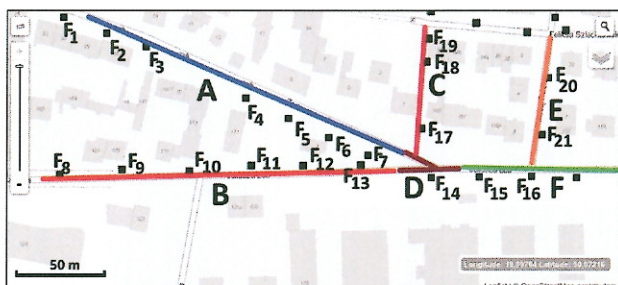
Na Rysunku 1 przedstawiono diagram ilustrujący problematykę badawczą podzieloną na bloki tematyczne, wraz z odniesieniami do najważniejszych publikacji. Blok *Zastosowania modelu* odpowiada wszelkim pracom o charakterze badań nad zastosowaniami (również praktycznymi) osiągniętych wyników, których autorem jest habilitant. Diagram ilustruje także związki pomiędzy poszczególnymi podtematami badawczymi¹.

(c) **Omówienie celu naukowego** ww. prac i osiągniętych wyników wraz z omówieniem ich ewentualnego wykorzystania.

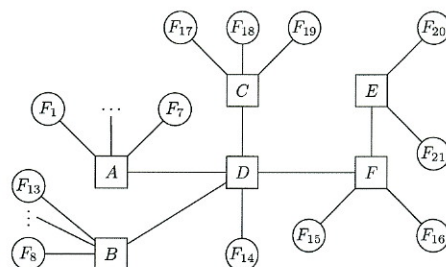
A. Cel naukowy prac i osiągnięte wyniki - streszczenie

Po uzyskaniu doktoratu habilitant podjął tematykę badawczą związaną z rozproszonymi grafami i transformacjami grafowymi. Prace te, mające początkowo charakter badań teoretycznych, miały na celu stworzenie środowiska obliczeniowego dla obliczeń rozproszonych, prowadzonych z wykorzystaniem modeli grafowych. Należy bowiem zauważyć, że opracowanie efektywnych algorytmów rozwiązujących szereg problemów o dużej złożoności, zarówno w sensie rozmiaru, jak i własności strukturalnych natrafia na dwie bariery. Są nimi: brak notacji formalnej umożliwiającej opis projektowanego systemu oraz niską efektywność obliczeniową środowisk bazujących na istniejących podejściach obliczeniowych.

Głównymi celami naukowymi badań prowadzonych przez habilitanta były zatem: (1) stworzenie efektywnego środowiska obliczeniowego opartego na rozproszonej reprezentacji grafowej [A2] oraz spójnego modelu dla obliczeń równoległych (MOR), prowadzonych w tym środowisku. Należy nadmienić, że efektywność odnosi się w tym kontekście do czasowej złożoności obliczeniowej algorytmów opartych na tym modelu, pozwalającej na ich praktyczne zastosowanie. Akceptowalna w tym kontekście jest złożoność wielomianowa; (2) opracowanie formalnej reprezentacji grafowej, modelującej środowisko przestrzenne, będącej jednocześnie częścią MOR. Cel ten związany był z planowanym obszarem zastosowań całego formalizmu (zob. Rysunek 2) [A4,A6,A7,A9].



(a) Mapka ilustrująca rozmieszczenie punktów świetlnych. A, B, C, D, E, F oznaczają wyodrębnione pola obliczeniowe, F_1, F_2, \dots, F_{21} reprezentują punkty świetlne (lampy)



(b) Graf reprezentujący scenę przedstawioną na Rysunku 2a. Węzły oznaczone kwadratami odpowiadają polom obliczeniowym, zaś okrągłe punktom świetlnym

Rysunek 2: Przykładowy fragment miasta i uproszczony model grafowy reprezentujący odpowiadające mu pola obliczeniowe (dla obliczeń fotometrycznych). Graf taki zostaje rozproszony w modelu grafów ciętych celem zrównoleżenia przetwarzania

Kluczowym postulatem dotyczącym tworzonego modelu obliczeń równoległych (dotyczy celu (1)) było zapewnienie jego uniwersalności, zarówno w zakresie akceptowalnych typów grafów, co pozwoliłoby na stosowanie go w problemach obliczeniowych, cechujących się różnym stopniem strukturalnej złożoności, jak i rodzajów transformacji grafowych, które w środowisku rozproszonym, być może heterogenicznym, miałyby być wykonywane (zob. [B3,B9]). Jako pożądany wynik przyjęto możliwość używania, w ramach stworzonego formalizmu, transformacji algebraicznych (*single pushout*, *double pushout*) i algorytmicznych (np. ETPL(k) [C1] lub ogólniej, NLC [C2]) i to w taki sposób, aby produkcje gramatyki działające na grafie scentralizowanym dziedziczone były bez żadnych modyfikacji przez grafy reprezentacji rozproszonej, stanowiącej integralny składnik formalizmu [A9]. Podejściem, które przyjęto dla realizacji obliczeń równoległych są obliczenia agentowe, w których na wiedzę agenta składa się zarówno struktura grafu, jak i istniejące reguły transformacyjne (gramatyki grafowe) [A1,B2,B3,B5,B9].

Trzyletnie prace badawcze zaowocowały opracowaniem modelu **replikowanych grafów komplementarnych** (ang.: *Replicated Complementary Graphs* – w skrócie: RCG) [A1] i koncepcji rozproszonych transformacji grafowych wykonywanych w tym środowisku. Zbudowany został również model **systemów wieloagentowych**, wykonujących te operacje.

¹Uwaga: referencje Axx odnoszą się do prac stanowiących wykazywane osiągnięcie, natomiast Bxx są referencjami do publikacji zamieszczonych w Dodatku, znajdującym się na końcu autoreferatu.

AGodny

W dalszym okresie kontynuowane były prace teoretyczne, ale także badania nad zastosowaniami osiągniętych już wyników. W szczególności dotyczy to międzynarodowego grantu *Products and Services of a Living Smart Energy City Lab*, rozpoczętego w 2011 roku. Jego celem było zwiększenie efektywności energetycznej instalacji oświetlenia zewnętrznego.

Prace teoretyczne przyniosły koncepcję **grafów ciętych** (ang.: *slashed graphs*) będącą indywidualnym wkładem habilitanta w teorię rozproszonych transformacji grafowych [A2,B5]. Jej wykorzystanie w znacznym stopniu redukuje złożoność obliczeniową operacji rozproszonych transformacji grafowych (mierzoną liczbą komunikatów przesyłanych pomiędzy agentami), umożliwiając praktyczne zastosowanie modelu, omówione w dalszej części.

Oprócz tego habilitant stworzył **hierarchiczny hipergrafowy model przestrzeni trójwymiarowej** [A3], będący oryginalnym uogólnieniem koncepcji hierarchicznego hipergrafu sąsiedztwa ścian [C3]. Jedną z istotnych własności modelu jest istnienie wprowadzonych przez habilitanta, komplementarnych do siebie operacji dekompozycji (analizy) i agregacji (syntezy) hipergrafów. Operacje te odpowiadają odpowiednio zastosowaniu produkcji pewnej gramatyki hipergrafowej oraz analizie składniowej (według tejże gramatyki) pewnych fragmentów hipergrafu. Transformacja syntezy, co zostanie pokazane w dalszej części, umożliwia przejście z modelu hipergrafowego do grafowego. To z kolei pozwala na wykorzystanie formalizmu grafów ciętych dla takiej w pełni zagregowanej struktury. Punkt ten jest zwornikiem głównej osi badawczej z nurtem aplikacyjnym prowadzonych prac. Połączenie stworzonych przez habilitanta koncepcji hierarchicznej reprezentacji hipergrafowej przestrzeni 3D oraz reprezentacji grafów ciętych umożliwiło znaczną redukcję czasowej złożoności zadań obliczeniowych [A10]. Zdecydowało to o zastosowaniu formalizmu w zagadnieniach projektowania architektonicznego (systemy typu CAD) [A9], a także w problemach projektowania oświetlenia [A5,A8], co przyniosło wymierne efekty praktyczne w postaci wdrożonych systemów oświetleniowych (zob. punkt (d)).

B. Osiągnięte wyniki - omówienie

W sekcjach B.1 oraz B.2 przedstawiono dwa modele rozproszonych reprezentacji grafowych: model grafów komplementarnych oraz model grafów ciętych. W sekcji B.3 pomieszczono zarys hierarchicznej hipergrafowej reprezentacji obiektów trójwymiarowych. Sekcja B.4 zawiera charakterystykę systemu agentowego realizującego obliczenia równoległe w środowisku rozproszonym. Wreszcie, w sekcji B.5 zamieszczono opis **zastosowania rozproszonych modeli i transformacji grafowych opartych o reprezentację grafów ciętych (slashed graphs) w problemie projektowania i sterowania oświetleniem zewnętrznym**, oparty o rzeczywiste prace badawczo-wdrożeniowe.

B.1. Grafy komplementarne [A1]

Etapem poprzedzającym definicję grafów komplementarnych jest wprowadzenie pojęcia tzw. *IE-grafu* (por. [A1]), stanowiącego strukturę bazową dla przedstawianej dalej operacji rozpraszania.

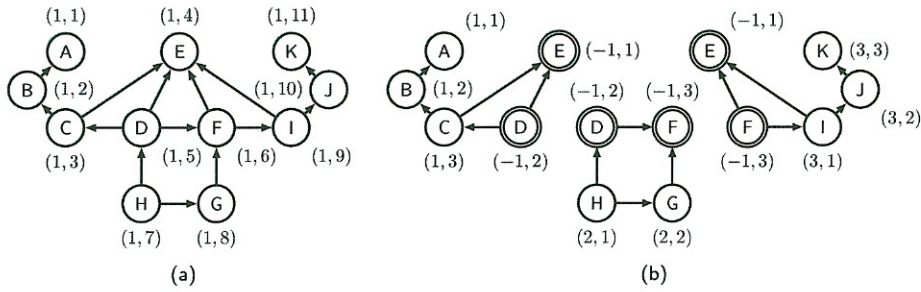
Definicja 1 *IE-grafem* nad zbiorami Σ i Γ nazywamy piątkę postaci $G = (V, E, \Sigma, \Gamma, \delta)$, gdzie:

- V jest skończonym, niepustym zbiorem węzłów grafu, do których przypisano unikalne indeksy należące do uporządkowanego zbioru I ; porządek w zbiorze I definiuje porządek w zbiorze V ,
- $E \subset V \times \Gamma \times V$ jest zbiorem krawędzi grafu,
- Σ jest zbiorem etykiet węzłów,
- $\Gamma = \{\gamma_1, \gamma_2, \dots, \gamma_n : \gamma_1 \leq \dots \leq \gamma_n\}$ jest skończonym, niepustym i uporządkowanym zbiorem etykiet krawędzi,
- $\delta : V \rightarrow \Gamma$ jest funkcją etykietującą węzły.

Rodzinę *IE-grafów* oznaczymy jako \mathcal{G}_{IE} .

Niech $G_i = (V_i, E_i, \Sigma_i, \Gamma_i, \delta_i) \in \mathcal{G}_{IE}$. Wprowadza się następującą notację:

- $\text{Replicas}(G_i)$ oznacza zbiór wszystkich węzłów grafu G_i , dla których istnieją repliki w innych grafach $G_{j \neq i}$ (tzw. węzły **brzegowe**).
- $\text{Private}(G_i)$ oznacza zbiór wszystkich węzłów grafu G_i , które nie posiadają replik w innych grafach (tzw. węzły **wewnętrzne**).
- $(v, w)_{G_i}$ oznacza parę (v, w) , taką że: $(v, \gamma, w) \in E_i$ lub $(w, \gamma, v) \in E_i$, dla pewnego $\gamma \in \Gamma_i$.
- $(v, w)_{G_i}^+$ oznacza ciąg węzłów $u_1, u_2, \dots, u_k \in V_i$, taki że $v = u_1, w = u_k, u_m \neq u_n$ dla $m \neq n$ i $\exists (u_m, u_{m+1})_{G_i}$ dla $m < k$; ciąg (u_1, u_2, \dots, u_k) nazwiemy uogólnionym acyklicznym połączeniem między węzłami v i w .



Rysunek 3: (a) Graf G w postaci scentralizowanej (b) Postać komplementarna grafu G , $\{G_1, G_2, G_3\}$

- $\text{Path}(G_i, v, w)$ jest zbiorem wszystkich uogólnionych acyklicznych połączeń między węzłami v i w ; zbiór węzłów należących do $\text{Path}(G_i, v, w)$ będzie oznaczony jako $\text{PNodes}(G_i, v, w)$.

Rysunek 3(a) przedstawia graf G , w którym $\text{PNodes}(G, C, I) = \{C, D, E, F, G, H, I\}$.

Definicja 2 Zbiór grafów $G_i = (V_i, E_i, \varphi_i) \in \mathcal{G}_{IE}$, $i = 1, 2, \dots, k$, jest **replikowaną i komplementarną postacią grafu** G wtedy i tylko wtedy, gdy istnieje zbiór iniektywnych homomorfizmów $s_i : G_i \rightarrow G$, takich że:

- $\bigcup_{i=1, \dots, k} s_i(G_i) = G$,
- $\forall i, j \in \{1, \dots, k\} : s_i(V_i) \cap s_j(V_j) = s_i(\text{Replicas}(G_i)) \cap s_j(\text{Replicas}(G_j))$,
- $\forall i \in \{1, \dots, k\} \text{Border}(G_i) = \{v \in \text{Replicas}(G_i) : \text{istnieje } (v, w)_{G_i} \text{ dla pewnego } w \in \text{Private}(G_i)\}$; $\text{Border}(G_i)$ jest nazwane zbiorem **węzłów brzegowych** grafu G_i ,
- $\forall w \in \text{Private}(G_i) \forall v \in \text{Private}(G_j) \forall u_1, u_2, \dots, u_k \in \text{PNodes}(G_i, w, v) \Rightarrow \exists b \in \text{Border}(G_i) \exists 1 \leq p \leq k : s_i(b) = u_p$.

Graf G_i nazywamy **replikowanym grafem komplementarnym** (ang. *replicated complementary graph – RCG*).

Z każdym replikowanym grafem komplementarnym związany jest osobny agent obliczeniowy, realizujący transformacje grafowe [A1]. Mogą mieć one bądź charakter wstępnego przygotowania rozproszonego środowiska grafowego do obliczeń (np. przez odpowiednie partycjonowanie), bądź mogą stanowić zasadniczą część obliczeń agentowych. W każdym z powyższych przypadków, ich wykonanie wiąże się z realizacją ciągu procedur pomocniczych. Wykonywane są one w procesie rozpraszania lub przygotowują one środowisko grafowe do lokalnego zastosowania danej produkcji (uzyskanie przez agenta wyłącznego dostępu do lewej strony planowanej produkcji i jej kontekstu). Procedury te zebrano poniżej.

SplitGraph(G_{01}, V_H) - G_{01} jest danym grafem wejściowym oraz $V_H \subset V(G_{01})$ pewnym podzbiorem jego węzłów. Operacja prowadzi do utworzenia dwóch replikowanych grafów komplementarnych, G_{10} i G_{11} . G_{10} oraz G_{11} są utworzone odpowiednio dla zbiorów V_H i $V(G_{01}) \setminus V_H$ zgodnie z regułą zawarta w pracy [B1] (zob. dowód Twierdzenia 1). Dla nowo powstałego grafu G_{10} tworzony jest osobny agent, natomiast agent wykonujący operację **SplitGraph**(G_{01}, V_H) przejmuje graf G_{11} zastępujący G_{01} .

JoinGraphs(DEL) operacja polega na wysłaniu przez agenta A do wszystkich pozostałych agentów żądania nadesłania kopii ich grafów komplementarnych, które następnie scalane są przez A , w trybie transakcyjnym, do scentralizowanego grafu G . Jeśli parametr DEL = true wówczas agenci po odesłaniu żądanej kopii kończą swoje działanie.

Expand(v, i, COND) - dla każdej repliki u danego węzła $v \in \text{Replicas}(G_i)$, wszyscy sąsiedzi u spełniający dany warunek COND, są replikowani, a te nowo utworzone repliki zostają dołączone do grafu to G_i razem z krawędziami łączącymi je z węzłem u . Warunek COND pozwala ograniczyć rozrost grafu G_i poprzez pobieranie jedynie tych węzłów, których etykiety, atrybuty lub inne własności spełniają warunek COND.

Incorporate(v, i) - operacja ta powoduje, że dla danego węzła $v \in \text{Replicas}(G_i)$ przesunięta zostaje granica pomiędzy grafem komplementarnym G_i a jego grafem sąsiednim i odbywa się to w sposób następujący. Niech u oznacza pewną replikę węzła v , wówczas wszystkie węzły sąsiednie do u (istniejące w innych grafach komplementarnych) zostają zreplikowane do G_i razem z krawędziami incydentnymi z węzłem u . Następnie, wszystkie repliki v , oprócz samego v (a więc i węzeł u) zostają usunięte ze swoich grafów komplementarnych, a węzeł v staje się węzłem wewnętrznym.

Handwritten signature

Neighborhood (v, G, k) - dla wężła $v \in V(G_i)$ procedura zwraca graf B , taki że $V(B) = \{w : \exists p \in \text{Path}(G, v, w) \text{ i liczba wężł\u00f3w w } p \text{ jest nie wi\u0119ksza ni\u017c } k\}$. Wyznaczenie zbioru B wymaga rekursywnej wsp\u00f3lpracy agent\u00f3w obliczeniowych.

Jak wykazano w pracy [B1], wszystkie powy\u017csze operacje wykonywane w \u015brodowisku rozproszonym na postaci komplementarnej grafu G , posiadaj\u0105 wielomianow\u0105 z\u0142o\u017cono\u015b\u0107 obliczeniow\u0105, mierzon\u0105 ilo\u015bci\u0105 komunikat\u00f3w przesy\u0142anych pomi\u0119dzy agentami.

W\u0105жным krokiem w celu realizacji rozproszonych transformacji grafowych jest synchronizacja dzia\u0142a\u0144 podejmowanych przez agent\u00f3w w toku stosowania tych transformacji. W podej\u015bciu RCG stosowana jest synchronizacja niejawn\u0105 (ang.: *implicit synchronization*), zdejmuj\u0105ca z projektanta konieczno\u015b\u0107 podejmowania dzia\u0142a\u0144 algorytmicznych, zmierzaj\u0105cych do zapewnienia synchronizacji. Polega ona na przesuni\u0119ciu granicy mi\u0119dzy grafami komplementarnymi (za pomoc\u0105 procedury Incorporate) tak, aby mo\u017cliwe by\u0142o lokalne wykonanie produkcji danej gramatyki grafowej [A1,B2].

Istotn\u0105 cech\u0105 reprezentacji RCG jest mo\u017cliwo\u015b\u0107 replikowania wybranych informacji (fragment\u00f3w graf\u00f3w komplementarnych) w systemie rozproszonym, co stanowi u\u017cyteczn\u0105 w\u0142a\u015bno\u015b\u0107 w przypadku zastosowania modelu do system\u00f3w typu CAD (poza [A9] analizowane tak\u017ce w [B10]). Istnieje jednak klasa zada\u0144 obliczeniowych, w kt\u00f3rej replikacja taka ogranicza si\u0119 jedynie do w\u0119z\u0142\u00f3w brzegowych. Dla problem\u00f3w tej klasy habilitant stworzy\u0142 reprezentacj\u0119 graf\u00f3w ci\u0119tych [A2,B5], om\u00f3wion\u0105 ni\u017cej.

B.2. Post\u0105c ci\u0119ta grafu [A2]

Kluczowym elementem modelu obliczeniowego, decyduj\u0105cym o jego przydatno\u015bci w praktycznych zastosowaniach, jest stworzona przez habilitanta koncepcja reprezentacji ci\u0119tej grafu, om\u00f3wiona poni\u017cej.

Niech \mathcal{G} oznacza rodzin\u0119 graf\u00f3w skierowanych postaci $G = (V, E, lab)$, gdzie V jest niepustym zbiorem w\u0119z\u0142\u00f3w, $E \subseteq V \times (\mathcal{L}_E \times 2^{A_E}) \times V$ jest zbiorem kraw\u0119dzi skierowanych, $lab : V \rightarrow \mathcal{L}_V$ jest funkcj\u0105 etykietuj\u0105c\u0105 w\u0119z\u0142y, $\mathcal{L}_E, \mathcal{L}_V$ oznaczaj\u0105 odpowiednio zbiory etykiet kraw\u0119dzi i w\u0119z\u0142\u00f3w, natomiast A_E jest zbiorem atrybut\u00f3w kraw\u0119dzi.

Definicja 3 Post\u0105c ci\u0119ta grafu. Niech $G = (V, E, lab) \in \mathcal{G}$ oraz niech $\{G_i\}_{i \in I}$ b\u0119dzie zbiorem graf\u00f3w, takim \u017ce:

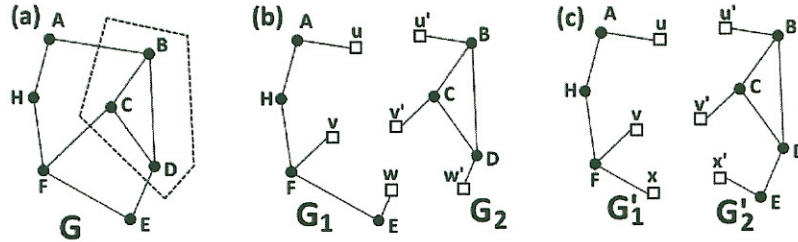
- I jest sko\u0144czonym, niepustym zbiorem indeks\u00f3w,
- $G_i = (V_i, E_i, lab_i) \in \mathcal{G}$ i $V_i = C_i \cup D_i, C_i \cap D_i = \emptyset$, gdzie C_i jest zbiorem **w\u0119z\u0142\u00f3w wewn\u0119trznych**, D_i oznacza zbi\u00f3r **w\u0119z\u0142\u00f3w atrapowych** i $lab_i \equiv lab|_{V_i}$,
- $\bigcup_i C_i = V$, gdzie C_i, C_j s\u0105 parami roz\u0142\u0105czne,
- $\forall v \in D_i \exists ! v' \in D_j (i \neq j)$, taki \u017ce v' jest replik\u0105 w\u0119z\u0142a v ; $\forall v \in D_i \deg(v) = 1$,
- $\forall e \in E_i : e$ jest incydentna z co najwy\u017cej jednym w\u0119z\u0142em atrapowym.

Kraw\u0119dz incydentna z w\u0119z\u0142em atrapowym nazywana jest **kraw\u0119dzi\u0105 brzegow\u0105**. Zbi\u00f3r wszystkich kraw\u0119dzi brzegowych w grafie G_i oznaczony b\u0119dzie przez E_i^b . Zbi\u00f3r $E_i^c = E_i - E_i^b$ nazywa si\u0119 zbiorem **kraw\u0119dzi wewn\u0119trznych** grafu G_i . Oznaczmy dla wi\u0119kszej przejrzysto\u015bci: $M = \mathcal{L}_E \times 2^{A_E}$, wtedy zbi\u00f3r $\{G_i\}$ zdefiniowany powy\u017cej stanowi **posta\u0107 (reprezentacj\u0119) ci\u0119t\u0105 grafu** G (ozn. \mathcal{G}), wtedy i tylko wtedy gdy spe\u0142nione s\u0105 nast\u0119puj\u0105ce warunki:

1. $\forall G_i^c = (C_i, E_i^c, lab_i|_{C_i}), \exists H_i \subset G : H_i \cong G_i^c$ (H_i i G_i^c s\u0105 sobie r\u00f3wne z dok\u0142adno\u015bci\u0105 do izomorfizmu α) i H_i, H_j s\u0105 roz\u0142\u0105czne dla $i \neq j$.
2. Istnieje bijekcja $f : M^2 \rightarrow M$ taka, \u017ce dla ka\u017cdej pary kraw\u0119dzi brzegowych $(e, e') \in E_i^b \times E_j^b (i \neq j)$, takich \u017ce:
 - (a) $e = (x_c, m, v) \in C_i \times M \times D_i, e' = (v', m', y_c) \in D_j \times M \times C_j$,
 - (b) v' jest replik\u0105 v ,
 istnieje dok\u0142adnie jedna kraw\u0119dz $e_{ij} = (x, m_e, y) \in E$ taka, \u017ce $x_c = \alpha(x), y_c = \alpha(y)$ i $f(m, m') = m_e$.
 e_{ij} jest okre\u015blana jako **kraw\u0119dz ci\u0119ta**, skojarzona z w\u0119z\u0142ami atrapowymi v, v' .
3. $\forall e = (x, m, y) \in E : (i) \exists ! e_c \in E_i^c$ dla pewnego i , taka, \u017ce $e_c = \alpha(e)$ lub (ii) $\exists ! (v, v') \in D_i \times D_j$ dla pewnych i, j , takich \u017ce e jest kraw\u0119dzi\u0105 ci\u0119t\u0105 skojarzon\u0105 z v i v' .

Graf $G_i \in \mathcal{G}$ jest okre\u015blany jako **sk\u0142adowa ci\u0119ta** G .

Rysunek 4 przedstawia przyk\u0142adowy graf (a) i jego post\u0105c ci\u0119t\u0105 (b).



Rysunek 4: (a) Przykładowy graf G : przerywana linia oznacza granicę planowanego podziału (b) Postać cięta $\mathcal{G} = \{G_1, G_2\}$ (c) Postać cięta po inkorporacji krawędzi brzegowej $e_{w'}$ do składowej G_2

W praktycznych zastosowaniach postaci ciętej istotnymi operacjami są: podział grafu (Split), scalanie grafów (Merge) oraz operacja tzw. inkorporacji (Incorporate), czyli przesunięcia granicy podziału między dwiema składowymi ciętymi (zob. Rysunek 4(b) i (c)) powiązanych poprzez pewną parę (pary) węzłów atrapowych, będących wzajemnie replikami. W dalszej części przedstawiono pseudokod odpowiadający procedurom Split oraz Merge, a także opis przebiegu procedury Incorporate.

Algorytm 1: Algorytm Split(G, V_c)

Wejście: $G = (C \cup D, E)$ – graf wejściowy, $V_c \subset C$ – pewien podzbiór zbioru węzłów wewnętrznych

Wyjście: $\mathcal{G} = \{G_1, G_2\}$, gdzie $G_i = (C_i \cup D_i, E_i)$

```

1 begin
2    $C_1 \leftarrow V_c$ ;
3    $D_1 \leftarrow$  wszystkie węzły atrapowe z  $D$ , sąsiednie do  $V_c$  w  $G$ ;
4    $E_1 \leftarrow$  wszystkie krawędzie łączące węzły z  $C_1 \cup D_1$  w  $G$ ;
5    $C_2 \leftarrow V - V_c$ ;
6    $D_2 \leftarrow$  wszystkie węzły atrapowe z  $D$ , sąsiednie do  $C - V_c$  w  $G$ ;
7    $E_2 \leftarrow$  wszystkie krawędzie łączące węzły z  $C_2 \cup D_2$  w  $G$ ;
8    $E_{conn} \leftarrow$  wszystkie krawędzie  $E$  łączące  $V_c$  i  $C - V_c$  w  $G$ ;
9   foreach  $e = (x, m_e, y) \in E_{conn}$  do
10    Utwórz węzeł atrapowy  $v$  oraz jego replikę  $v'$ ;
11     $(m, m') \leftarrow f^{-1}(m_e)$ ;
12     $e_v \leftarrow (x, m, v)$ ,  $e_{v'} \leftarrow (v', m', y)$ ;
13     $D_1 \leftarrow D_1 \cup \{v\}$ ;
14     $E_1 \leftarrow E_1 \cup \{e_v\}$ ;
15     $D_2 \leftarrow D_2 \cup \{v'\}$ ;
16     $E_2 \leftarrow E_2 \cup \{e_{v'}\}$ ;
17 return  $\{G_1, G_2\}$ ;

```

Czasowa złożoność obliczeniowa algorytmu wynosi $\mathcal{O}(|E|) = \mathcal{O}(|C \cup D|^2)$.

Algorytm 2: Algorytm Merge(G_i, G_j)

Wejście: G_i, G_j – składowe cięte postaci $G_i = (C_i \cup D_i, E_i)$

Wyjście: G_{ij} – graf uzyskany w wyniku połączenia grafów wejściowych

```

1 begin
2    $D_i^{common} \leftarrow \{v \in D_i : \exists v' \in D_j : v' \text{ jest repliką } v\}$ ;
3    $D_j^{common} \leftarrow$  repliki węzłów z  $D_i^{common}$  w  $G_j$ ;
4    $C_{ij} \leftarrow C_i \cup C_j$ ;
5    $D_{ij} \leftarrow (D_i - D_i^{common}) \cup (D_j - D_j^{common})$ ;
6    $E_{ij} \leftarrow (E_i - \{ \text{krawędzie } G_i \text{ incydentne z } D_i^{common} \}) \cup (E_j - \{ \text{krawędzie } G_j \text{ incydentne z } D_j^{common} \})$ ;
7   for  $e_v = (x, m, v) \in E_i, e_{v'} = (v', m', y) \in E_j$ , gdzie  $v'$  jest repliką  $v$  do
8      $m_e \leftarrow f(m, m')$ ;
9      $E_{ij} \leftarrow E_{ij} \cup \{(x, m_e, y)\}$ ;
10 return  $G_{ij} = (C_{ij} \cup D_{ij}, E_{ij})$ 

```

Czasowa złożoność obliczeniowa algorytmu wynosi $\mathcal{O}(|E|) = \mathcal{O}(|C \cup D|^2)$.

algorytm

Procedura $\text{Incorporate}(G_0, e_v)$, będąca atomową operacją w grafowym środowisku rozproszonym, ma przebieg jedno-fazowy. Założmy, że v' jest repliką wężła v a c jest węzłem wewnętrznym sąsiadującym z v' . Realizacja procedury $\text{Incorporate}(G_0, e_v)$ rozpoczynana jest przez agenta inicjującego A_0 , który wysyła żądanie commit do agenta A_1 zarządzającego grafem zawierającym c . A_1 może albo dostarczyć żądany wierzchołek c wraz z sąsiedztwem (jeśli takie istnieje), albo odrzucić żądanie wysyłając odpowiedź locked , gdy jeden z elementów tego sąsiedztwa został wcześniej zablokowany przez innego agenta inicjującego. Gdy A_0 otrzymuje żądany podgraf (zawierający węzeł c), wtedy wysyła żądanie aktualizujące update do wszystkich agentów posiadających repliki węzłów atrapowych, przyłączonych właśnie do G_0 . Żądania te odświeżają informacje dotyczące lokalizacji replik dodanych do G_0 . Należy zwrócić uwagę, że adresaci komunikatów update są znani, ponieważ ich referencje zawarte są w węzłach atrapowych. Jeśli odpowiedź od agenta A_1 jest komunikat locked , wówczas A_0 ponawia operację z losowym opóźnieniem.

Złożoność procedury Incorporate określona jest następującymi wielkościami. Ilość agentów uczestniczących w operacji inkorporacji wynosi $\mathcal{O}(2)$. Ilość węzłów/krawędzi blokowanych w jej trakcie wynosi $\mathcal{O}(d)$, gdzie d jest maksymalnym stopniem wężła w grafie. Ilość komunikatów aktualizujących, wysłanych po zakończeniu inkorporacji, wynosi $\mathcal{O}(d)$. Jak zatem widać, wszystkie trzy parametry są stałe względem ilości węzłów w grafie. **W szczególności zaś, złożoność operacji inkorporacji mierzona liczbą wysłanych komunikatów jest stała.** Postać cięta grafu stanowi zatem efektywniejszą reprezentację niż RCG, o ile moc opisowa gramatyki okaże się wystarczająca.

B.3. Modelowanie obiektów 3D [A3,A4,A8,A9]

Równie ważnym wątkiem badawczym w pracy habilitanta była hipergrafowa reprezentacja obiektów trójwymiarowych, przedstawiona w [A3]. Jej główne zręby stanowi koncepcja S-hipergrafu (def. 2) oraz grafu agregującego (def. 3) [A4].

Definicja 4 *S-hipergrafem* obiektu S jest ciąg postaci:

$$G_S = (N, A, H, \text{lab}_N, \text{lab}_A, \text{lab}_H, \text{att}_N, \text{att}_A, \text{att}_H),$$

gdzie N jest niepustym zbiorem węzłów, $A \subset P_2(N)$ jest zbiorem krawędzi, $H \subset \bigcup_{i>2} P_i(N)$ jest zbiorem hiperkrawędzi, $\text{lab}_\mu : \mu \rightarrow \mathcal{L}_\mu$ ($\mu = N, A, H$) są funkcjami etykietującymi odpowiednio węzły, krawędzie oraz hiperkrawędzie, z odpowiadającymi im zbiorami etykiet \mathcal{L}_μ ; $\text{att}_\mu : \mu \rightarrow 2^{A_\mu}$ ($\mu = N, A, H$) są funkcjami atrybutującymi odpowiednio węzły, krawędzie oraz hiperkrawędzie, przy czym wartością funkcji att_μ jest pewien podzbiór zbioru atrybutów (być może pusty).

Istotnym elementem modelu są funkcje atrybutujące (att), które zwiększają jego siłę ekspresji. I tak na przykład, formalizm S-hipergrafu pozwala na przedstawienie kuli (Rysunek 5), walca, czy innych nietrywialnych brył (rozmaitości), co ilustruje poniższy przykład.

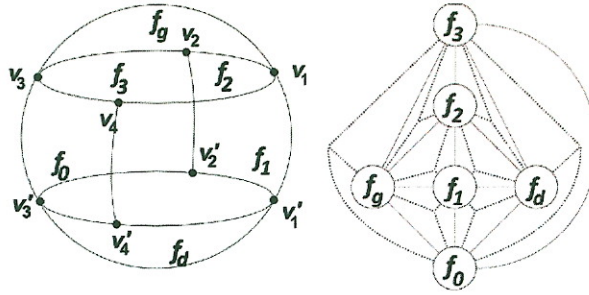
Przykład. Rozważmy kulę S o promieniu R , przedstawioną na Rysunku 5. Dla wygody opisu zakładamy, że jej środek pokrywa się z początkiem układu współrzędnych. Na kuli S znajduje się osiem „wierzchołków”: $v_1, \dots, v_4, v'_1, \dots, v'_4$ o współrzędnych kartezjańskich:

$$\begin{aligned} \mathbf{r}(v_1) &= (r_h, 0, h), & \mathbf{r}(v'_1) &= (r_h, 0, -h), \\ \mathbf{r}(v_2) &= (0, r_h, h), & \mathbf{r}(v'_2) &= (0, r_h, -h), \\ \mathbf{r}(v_3) &= (-r_h, 0, h), & \mathbf{r}(v'_3) &= (-r_h, 0, -h), \\ \mathbf{r}(v_4) &= (0, -r_h, h), & \mathbf{r}(v'_4) &= (0, -r_h, -h), \end{aligned}$$

gdzie $0 < h < R$ oraz $r_h = \sqrt{R^2 - h^2}$. „Ściany” wyodrębnione w sposób arbitralny ze sfery, oznaczono jako $f_0, \dots, f_3, f_g, f_d$. S posiada następującą reprezentację hipergrafową: $G_S = (V, A, H, \{\text{lab}_\nu\}_{\nu=V,A,H}, \{\text{att}_\nu\}_{\nu=V,A,H})$, gdzie:

$$\begin{aligned} V &= \{f_0, \dots, f_3, f_g, f_d\}, \\ A &= \{\{f_0, f_1\}, \{f_1, f_2\}, \{f_2, f_3\}, \{f_3, f_0\}, \{f_0, f_g\}, \{f_1, f_g\}, \{f_2, f_g\}, \\ &\quad \{f_3, f_g\}, \{f_0, f_d\}, \{f_1, f_d\}, \{f_2, f_d\}, \{f_3, f_d\}\}, \\ H &= \{\{f_1, f_2, f_g\}, \{f_2, f_3, f_g\}, \{f_3, f_0, f_g\}, \{f_0, f_1, f_g\}, \{f_1, f_2, f_d\}, \\ &\quad \{f_2, f_3, f_d\}, \{f_3, f_0, f_d\}, \{f_0, f_1, f_d\}\}. \end{aligned}$$

Alamy



Rysunek 5: Kula jako modelowana bryła i jej hipergrafowa reprezentacja. Sztucznie wprowadzone „wierzchołki” oznaczono jako $v_1, \dots, v_4, v'_1, \dots, v'_4$. Wyodrębnione w oparciu o nie „ściany” jako $f_g, f_d, f_0, \dots, f_3$. Odpowiadają im węzły hipergrafu o tych samych oznaczeniach. Dla przejrzystości pominięto etykiety hiperkrawędzi.

Zakładamy, że funkcje atrybutujące przekazują charakterystyki geometryczne S , wyrażone we współrzędnych sferycznych:

$$\mathbf{r}(r, \theta, \varphi) = \begin{bmatrix} r \cos \theta \cos \varphi \\ r \cos \theta \sin \varphi \\ r \sin \theta \end{bmatrix}.$$

Funkcja atrybutująca dla wierzchołków grafu opisuje geometrię poszczególnych powierzchni:

$$att_V(f_g) = \{\mathbf{r} = \mathbf{r}(R, \theta, \varphi) : \theta \in (\theta_0, \frac{\pi}{2}), \varphi \in [0, 2\pi)\},$$

$$att_V(f_d) = \{\mathbf{r} = \mathbf{r}(R, \theta, \varphi) : \theta \in [-\frac{\pi}{2}, -\theta_0), \varphi \in [0, 2\pi)\},$$

$$att_V(f_i) = \{\mathbf{r} = \mathbf{r}(R, \theta, \varphi) : \theta \in [-\theta_0, \theta_0], \varphi \in [i\frac{\pi}{2}, (i+1)\frac{\pi}{2})\}, i = 0, 1, 2, 3.$$

Funkcja atrybutująca dla krawędzi z A charakteryzuje krzywizny krawędzi bryły:

$$att_A(\{f_i, f_{(i+1) \bmod 4}\}) = \{\mathbf{r} = \mathbf{r}(R, \theta, (i+1)\frac{\pi}{2}) : \theta \in [-\theta_0, \theta_0]\}, i = 0, 1, 2, 3,$$

$$att_A(\{f_i, f_g\}) = \{\mathbf{r} = \mathbf{r}(R, \theta_0, \varphi) : \varphi \in [i\frac{\pi}{2}, (i+1)\frac{\pi}{2})\}, i = 0, 1, 2, 3,$$

$$att_A(\{f_i, f_d\}) = \{\mathbf{r} = \mathbf{r}(R, -\theta_0, \varphi) : \varphi \in [i\frac{\pi}{2}, (i+1)\frac{\pi}{2})\}, i = 0, 1, 2, 3.$$

Funkcja atrybutująca dla hiperkrawędzi zwraca współrzędne reprezentowanych wierzchołków bryły S : $att_H(h_i) = \{\mathbf{r}(v_i)\}$.

Nie są określone żadne wymagania odnośnie funkcji etykietujących lab_V, lab_A, lab_H .

W celu rozszerzenia stosowalności modelu hipergrafowego na dowolny zbiór brył trójwymiarowych, a zarazem dla zapewnienia skalowalności takiej reprezentacji, habilitant powiązał koncepcję S-hipergrafu

$$G_S = (N, A, H, lab_N, lab_A, lab_H, att_N, att_A, att_H),$$

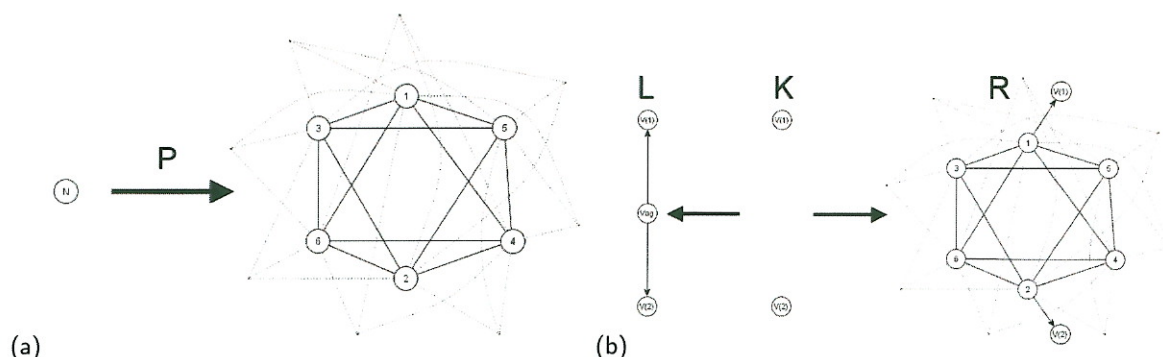
ze zwykłym grafem $G = (V, E, lab)$, który mógłby zostać rozproszony do postaci ciętej. W podejściu tym (def. 3), każdy węzeł ze zbioru V reprezentuje pewną bryłę ze zbioru $\{S_1, S_2, \dots, S_n\}$, a ściślej, istnieje bijekcja $f : V \rightarrow \{G_{S_1}, G_{S_2}, \dots, G_{S_n}\}$.

Definicja 5 Niech dany będzie graf $G = (V, E, lab) \in \mathcal{G}$. Załóżmy ponadto, że każdemu węzłowi $v \in V$ odpowiada pewien S-hipergraf G_S , natomiast każdej krawędzi $e = (u, m \in \mathcal{L}_E \times 2^{A_E}, w) \in E$, pewna relacja pomiędzy bryłami U i W , takimi że G_U i G_W są reprezentowane odpowiednio przez węzły u i v . Tak zdefiniowany graf G nazwiemy **grafem agregującym** dla ciągu brył S_1, S_2, \dots, S_n , natomiast $v \in V$ **węzłem agregującym**. Rodzinę grafów agregujących oznaczymy przez \mathcal{G}_A .

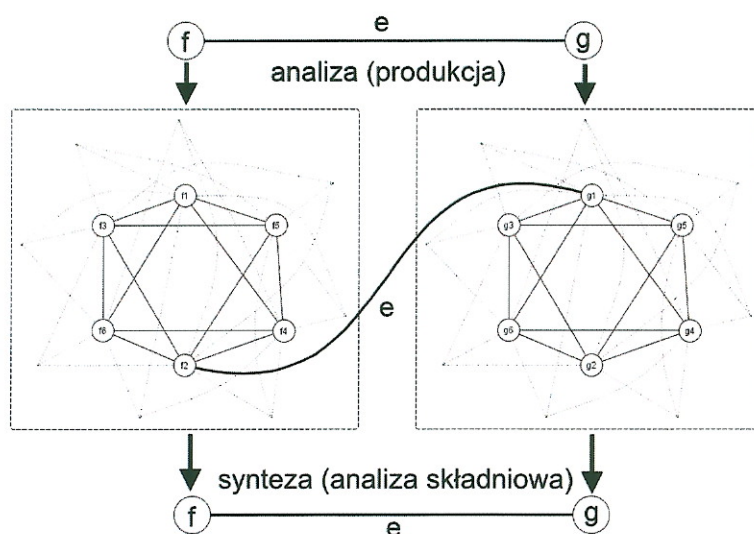
Analiza i synteza. Celem przejścia od górnego (węzły agregujące) do dolnego (hipergrafy) poziomu modelu hierarchicznego, stosuje się transformację analizy, realizowaną przez produkcję gramatyki hipergrafowej, która przeprowadza węzeł agregujący v w pewien odpowiadający mu hipergraf, $G_S : P : v \rightarrow G_S$ (por. Rysunek 6). Krawędź grafu agregującego incydentna wcześniej z węzłem v zostaje przyłączona do odpowiedniego węzła ze zbioru N , co specyfikuje transformacja osadzenia skojarzona z produkcją P . Należy zwrócić uwagę, że tak otrzymany hipergraf jest obiektem „hybrydowym” postaci

$$\hat{G} = (N, V, A, E, H, lab_{\{N, V, A, E, H\}}, att_{\{N, A, E, H\}}),$$

Alchimy



Rysunek 6: Produkcje transformujące węzeł agregujący do hipergrafu: (a) przypadek produkcji typu algorytmicznego, (b) produkcja typu *double pushout*



Rysunek 7: Transformacje analizy i syntezy. f i g są wierzchołkami grafu agregującego, natomiast e oznacza łączącą je krawędź.

gdzie zbiory są dziedziczone z poziomu dolnego (N, A, H , przy czym dopuszcza się tu, że $N = \emptyset$) oraz górnego (V) reprezentacji hierarchicznej, natomiast modyfikacji ulega zbiór E : $E \subseteq (V \cup N) \times (\mathcal{L}_E \times 2^{A_E}) \times (V \cup N)$. Analogicznie dzieje się z funkcjami etykietującymi i atrybutującymi, które dziedziczone są bez zmian, za wyjątkiem lab_E, att_E , które mogą ulec modyfikacjom związanym z produkcjami analizy. Rodzinę grafów \hat{G} oznaczmy przez $\hat{\mathcal{G}}$. Wówczas zachodzi własność $\mathcal{G}_A \subset \hat{\mathcal{G}}$, mówiąca, że graf agregowany jest szczególnym przypadkiem hipergrafu „hybrydowego”.

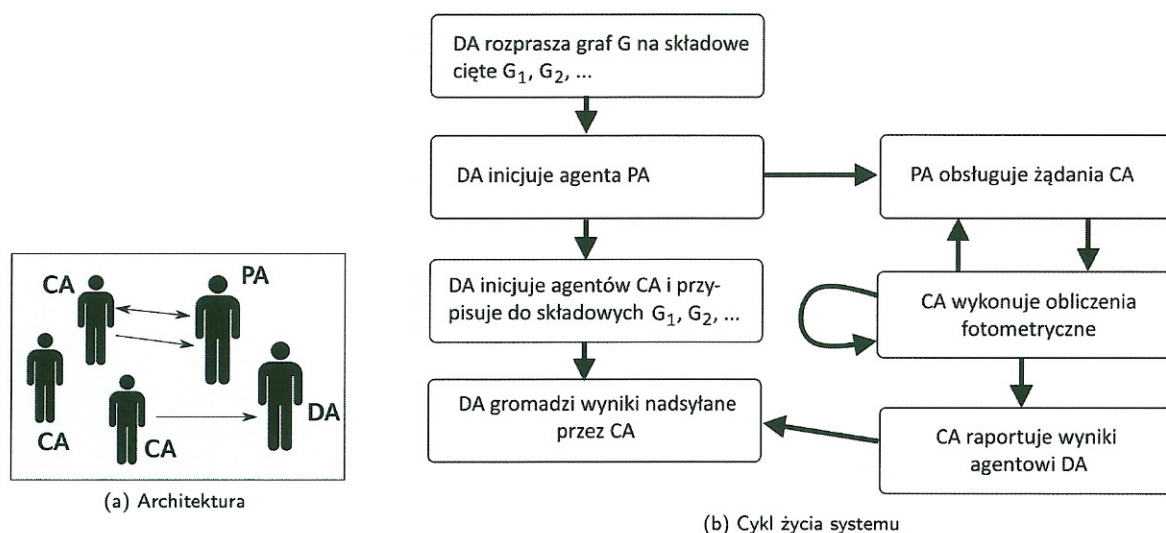
Transformacja odwrotna, synteza, a więc powrót do górnej warstwy modelu hierarchicznego, jest wykonywana poprzez analizę składniową grafu \hat{G} . Schemat operacji analizy i syntezy przedstawia Rysunek 7.

Kluczową własnością modelu stworzonego przez habilitanta jest fakt, iż dzięki zdefiniowanej wcześniej koncepcji grafów ciętych, możliwa jest dekompozycja scentralizowanego grafu agregującego, pozwalająca na równoległe lokalne wykonywanie zadań związanych z poszczególnymi hipergrafami (obiektami lub grupami obiektów fizycznych). A zatem, operacje analizy i syntezy przeprowadzane są równoległe i lokalnie na poziomie poszczególnych podzadań.

B.4. System agentowy [A1,A2,A4]

Habilitant przyjął, że wykorzystanie systemu agentowego będzie optymalne w rozważanym kontekście z uwagi na naturę problemu obliczeniowego, w którym proces optymalizacji odbywa się lokalnie, w sposób autonomiczny, z niewielkim udziałem danych pochodzących z sąsiednich podproblemów. Ponadto, w szeregu prac, habilitant sformułował podstawy

Alchimy



Rysunek 8: System agentowy realizujący projekt fotometryczny

formalne takiego modelu i udowodnił jego doskonałe własności w zakresie złożoności obliczeniowej. Jedną z jego zalet jest zdolność do samoorganizacji systemu agentowego pozwalającej na zrównoważenie rozmiaru zadań obliczeniowych w obrębie całego systemu agentowego (ang. *load balancing*). Proces ten, oparty o mechanizmy relaksacyjne, pozwala na osiągnięcie zrównoważenia na poziomie 91-95%, w zależności od średniej początkowej wielkości podzadania [B4]. Poniżej przedstawiono generyczną architekturę rozważanego systemu agentowego.

Architektura. Wyróżnia się trzy typy agentów:

1. agent dyspozytor (ang. *dispatcher agent* (DA)) – rolą DA, występującego zawsze w jednej instancji, jest tworzenie wszystkich innych agentów, zbieranie wyników oraz ewentualna obsługa innych zdarzeń (np. utrata agenta),
2. agent rejestrujący (ang. *pattern agent* (PA)) – rolą PA, występującego również w jednej instancji, jest z jednej strony wypełnianie tablicy asocjacyjnej postaci

[wzorzec danych wejściowych \rightarrow wynik],

zawierającej wszystkie przetwarzane dotychczas wzorce danych wejściowych dla problemów optymalizacyjnych, z drugiej zaś wyszukiwanie w niej wyników dla nadsyłanych wzorców (o ile takie istnieją w tablicy),

3. agent obliczeniowy (ang. *computing agent* (CA)) – wykonuje obliczenia na podległym mu podobzarsze i rejestruje u PA wzorce danych wejściowych, wraz z odpowiadającymi im wynikami.

W powyższym opisie nie dyskutuje się pewnych wtórnych kwestii, związanych z funkcjonowaniem systemu agentowego, jak na przykład istnienie zapasowych agentów DA czy PA, zapewniających odporność na uszkodzenia (ang. *fault tolerance*), czy też stosowanie wielu instancji agenta rejestrującego celem zmniejszenia czasu oczekiwania na odpowiedź przez agenta obliczeniowego.

Metoda powtórnego wykorzystania wyników. Dla usprawnienia działania systemu agentowego habilitant zastosował metodę powtórnego wykorzystania wyników [B7], będącą adaptacją metodologii DRY (*Don't Repeat Yourself*) [C4]. Oparta jest ona na obserwacji, że w pewnych problemach obliczeniowych o dużej skali, w których dane wejściowe można scharakteryzować za pomocą wzorców i dane te są powtarzalne, uprzednio obliczony wynik może być przy kolejnym wystąpieniu tego wzorca odczytany z pamięci asocjacyjnej a nie powtórnie wyliczany. Podejście takie skutkuje znaczną redukcją czasu obliczeń w przypadku dużej złożoności czasowej atomowego problemu obliczeniowego (tj. odpowiadającego pojedynczemu wzorcowi), która wzrasta razem z powtarzalnością wzorca.

Przykład. Przypuśćmy dla uproszczenia, że obliczenia fotometryczne wykonywane dla wyznaczenia optymalnej konfiguracji miejskiego systemu oświetleniowego realizowane są „ulica po ulicy” oraz, że wzorzec obliczeń (dane wejściowe) dla ulicy S ma postać

$$p(S) = (\text{wysokość słupa oświetleniowego, szerokość jezdni, średni odstęp między słupami}).$$

Wówczas obliczenie optymalnej konfiguracji dla rzędu lamp, skojarzonego z jezdnią S , rozpoczyna się od sprawdzenia, czy w pamięci asocjacyjnej (początkowo puste) nie występuje wzorec $p(S)$. Jeśli tak, wynik odczytywany jest z tablicy. W przeciwnym zaś razie, agent obliczeniowy wykonuje optymalizację dla wzorca $p(S)$ i otrzymany wynik rejestruje w pamięci asocjacyjnej PA. □

Pseudokody prezentujące działanie agenta obliczeniowego oraz agenta rejestrującego przedstawione są w Algorytmach 3 i 4.

Algorytm 3: Agent obliczeniowy – ComputeBehaviour(G)

input : G – przetwarzana składowa cięta
output: $T[G]$ – tablica wyników dla ulic reprezentowanych przez składową G

```

1 begin
2   const QuerySize;
3   Opróżnij kolejkę wiadomości;
4    $S \leftarrow$  lista ulic reprezentowanych przez składową  $G$ ;
5    $\Pi \leftarrow$  wzorce skojarzone z ulicami ze zbioru  $S$ ;
6    $P_1, P_2, \dots, P_k \leftarrow$  co najwyżej QuerySize-elementowe, parami
7   rozłączne listy, t.ż.  $\Pi = \bigcup_{i=1}^k P_i$ ;
8   foreach  $q \in \{P_1, \dots, P_k\}$  do
9     Wyślij do PA zapytanie QUERY( $q$ ) o wyniki skojarzone z wzorcami z  $q$ ;
10    if Przekroczony czas oczekiwania lub brak wyniku dla pewnych wzorców  $p$  then
11      Oblicz wyniki  $r$  dla wzorców  $p$ ;
12      Zarejestruj wyniki u PA: REGISTER( $[p, r]$ );
13  Skojarz wyniki z każdą z ulic z listy  $S$ ;
14  Raportuj wyniki końcowe do DA: REGISTER( $T[G]$ );
```

Algorytm 4: Agent rejestrujący – ServiceQueryBehaviour

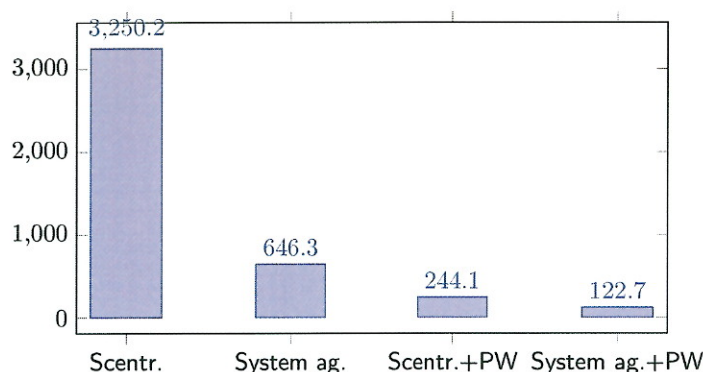
```

1 begin
2    $PV \leftarrow$  tablica asocjacyjna wzorec-wartość;
3   while Nie koniec do
4      $m \leftarrow$  receiveMessage();
5     if  $m$  jest żądaniem REGISTER then
6        $[(p, v)] \leftarrow$  odczytaj pary wzorec-wartość z wiadomości  $m$ ;
7        $PV \leftarrow PV \cup [(p, v)]$ ;
8     if  $m$  jest żądaniem QUERY then
9        $[p] \leftarrow$  lista wzorców odczytana z wiadomości  $m$ ;
10       $[r] \leftarrow$  wyniki odczytane z PV, skojarzone z  $[p]$ ;
11      Wyślij odpowiedź do CA: RESPONSE( $[r]$ );
```

Porównanie efektywności. Dla zbadania wpływu metody na czas trwania obliczeń, przeprowadzono szereg testów, w których graf $G = (V, E)$ reprezentujący fragment miasta (źródło: OpenStreetMap) przetwarzany był w następujących podejściach:

- sekwencyjnie,
- sekwencyjnie z powtórным wykorzystaniem wyników,
- równoległe, za pomocą systemu agentowego,
- równoległe, za pomocą systemu agentowego z powtórным wykorzystaniem wyników.

Graf G ($|V| = 1500$, $|E| = 2886$) rozproszony został przez DA na szereg podgrafów, $G_i = (V_i, E_i)$, takich że $|E_i| \leq E_{max} = 30$. Do każdego G_i przypisano osobnego agenta obliczeniowego, CA. Jego celem było znalezienie



Rysunek 9: Czasy wykonania obliczeń [sek.] dla różnych metod obliczeniowych. (Scentr. – obliczenia sekwencyjne na grafie scentralizowanym, System ag. – agentowe obliczenia równoległe, PW – metoda powtórnego wykorzystania wyników)

takiej konfiguracji instalacji oświetleniowej (a więc wysokości słupa, odstępów lamp, poziomu ściemnienia) dla danej ulicy bądź odcinka ulicy, która zapewniałaby spełnienie normy oświetlenia dla danej sytuacji, z drugiej zaś, minimalizację poboru mocy. Optymalizacja ta oparta była o *naiwny* schemat typu *brute force*, a dokładnie, sekwencję zagnieżdżonych pętli. System agentowy dla przeprowadzanych testów zaimplementowany został przez habilitanta z wykorzystaniem platformy JADE, w zgodności ze specyfikacją FIPA (Foundation for Intelligent Physical Agents). Wykres przedstawiony na Rysunku 9 ilustruje czasy wykonania dla poszczególnych metod obliczeniowych.

B.5. Studium przypadku zastosowania modelu obliczeniowego [A4–A8]

Pomyślnie praktyczne zastosowanie formalizmu grafów ciętych i analiza potencjalnych możliwości obliczeniowych tego podejścia umożliwiły **całkowitą zmianę paradygmatu projektowania instalacji oświetlenia zewnętrznego**. W swojej pracy opublikowanej na łamach czasopisma wydawanego przez *Illuminating Engineering Society Of North America* [A5], habilitant udowodnił, że porzucenie założenia o jednorodności układu drogowego i instalacji oświetleniowej (w obu przypadkach chodzi o symetrię translacyjną wzdłuż osi jezdni) na rzecz przyjęcia rzeczywistych, dokładnych współrzędnych, skutkuje polepszeniem efektywności energetycznej nawet o 15% w stosunku do nieoptymalizowanej instalacji. Podejście to, tzw. *customized design*, gwarantuje jednocześnie normatywny poziom oświetlenia. Dodatkowo, możliwe jest uwzględnienie światła odbitego od elewacji budynków czy pochodzącego z innych źródeł (tzw. *ambient light*). Tradycyjne podejście do projektowania oświetlenia, oparte na założeniu o jednorodności pozwalało na redukcję mocy instalacji na poziomie jedynie do 7%.

Jedynymi istotnymi problemami, jakie pojawią się w kontekście optymalizacji wielkoskalowej są: interferencja zadań obliczeniowych oraz problem stopu dla równoległego przetwarzania agentowego. Rozwiązanie tych problemów, oparte o zastosowanie koncepcji entropii systemu obliczeniowego zostało zaproponowane w pracy [A4].

Problem projektowania wielkoskalowych instalacji oświetleniowych stał się aktualny w ostatnich latach, wraz z coraz powszechniejszym wykorzystaniem technologii SSL (*Solid State Lighting*), a więc źródeł opartych o diody elektroluminescencyjne (potocznie tzw. lampy LED-owe). W ogólnym przypadku, powyższy problem polega na znalezieniu konfiguracji instalacji oświetleniowych zapewniających oświetlenie zgodne z obowiązującą normą (w Europie są to standardy EN 13201-2 oraz CIE 115) oraz wymagających minimalnego poboru mocy. Dotyczy to instalacji na obszarze całego miasta, a więc dziesiątków tysięcy opraw. Powyższe sformułowanie problemu należy uzupełnić dodatkowymi uwagami.

Uwaga 1. Do każdej ulicy (lub jej poszczególnych odcinków), skrzyżowania, ciągu pieszego itp. przypisana jest odrębnie tzw. klasa oświetleniowa, określająca wymagania oświetleniowe dla sytuacji drogowej danego rodzaju.

Uwaga 2. Zgodnie ze stosowaną praktyką, klasa oświetleniowa dla danego obszaru może ulec zmianie, np. w związku ze zmieniającym się natężeniem ruchu. Fakt ten stwarza możliwość zmniejszania natężenia światła (a więc i poboru energii) w określonych okresach w ciągu doby, np. w nocy, co znajduje swoje odzwierciedlenie w adaptacyjnych systemach sterowania oświetleniem (zob. Uwaga 4).

Uwaga 3. Klasyczne podejście do projektowania instalacji oświetleniowych, zakładające geometryczną jednorodność jezdni i rozmieszczenia opraw, zastąpione jest wspomnianym podejściem, przedstawionym przez habilitanta w pracy [A5]. W obliczeniach wykorzystuje się dokładne dane lokalizacyjne lamp oraz układu drogowego, dzięki

czemu, poprzez likwidację prześwietleń, uzyskuje się znaczącą redukcję wymaganej mocy. Podejście to powoduje jednak istotny wzrost złożoności obliczeniowej.

Uwaga 4. Dodatkowym środkiem pozwalającym na redukcję poboru energii jest wdrożenie systemu sterowania oświetleniem, opartego o koncepcję tzw. *profilu oświetleniowych* [A6,A7,B11]. Profilem oświetleniowym nazywa się predefiniowany zbiór nastaw dla określonej oprawy (grupy opraw), skojarzony z ustalonym stanem środowiska (natężenie ruchu, warunki atmosferyczne, obecność pieszych, obecność innych źródeł światła itp.). Sterowanie adaptacyjne oparte o koncepcję profili polega na dynamicznej zmianie profilu, a więc i nastaw, w zależności od warunków zewnętrznych, raportowanych z warstwy telemetrycznej (tj. przez sensory). Przygotowanie profili oświetleniowych wymaga jednakże wstępnego przeliczenia wszystkich sytuacji oświetleniowych dla poszczególnych stanów środowiska. Zwiększa to zatem złożoność obliczeniową zadania projektowania systemu.

O ile wykonanie projektu fotometrycznego (i optymalizacji nastaw) dla pojedynczej ulicy, przy założeniu jednorodności sytuacji oświetleniowej, jest wykonalne przy użyciu dostępnego oprogramowania, o tyle wielkoskalowa (tysiące opraw) optymalizacja projektu, szczególnie opartego o rzeczywiste dane geolokalizacyjne (Uwaga 3) jest niewykonalna w rozsądnym czasie przy użyciu istniejących aktualnie narzędzi. Przyczyną tego stanu rzeczy jest ich nieskalowalność. Jak wskazał habilitant w swoich pracach (np. [A8,A4]), środkiem zaradczym jest (i) sformalizowanie problemu optymalizacji projektu fotometrycznego tak, aby problem ten stał się zrozumiały dla systemu komputerowego, (ii) pełna automatyzacja procesu obliczeniowego, celem wykluczenia człowieka, stanowiącego jego wąskie gardło, (iii) zrównoleglenie procesu optymalizacji projektu fotometrycznego. Spełnienie wymienionych postulatów umożliwi stworzony przez habilitanta model hierarchicznej, rozproszonej reprezentacji grafowej, naszkicowany powyżej.

Został on wykorzystany w trakcie realizacji projektu *Pilotażowa modernizacja oświetlenia ulicznego Miasta Krakowa wraz z rozbudową warstwy telemetrycznej, stworzeniem systemu sterowania oraz budowa instalacji PV*, współfinansowanego przez Narodowy Fundusz Ochrony Środowiska i Gospodarki Wodnej, prowadzonego wspólnie z krakowskim Zarządem Infrastruktury Komunalnej i Transportu oraz firmą Schreder Polska (jako dostawcą systemu oświetleniowego). Celem operacyjnym projektu było poprawa efektywności energetycznej instalacji oświetleniowej poprzez wymianę ok. 4000 opraw sodowych na LED-owe oraz wdrożenie systemu inteligentnego sterowania. Podlegające wymianie oprawy zlokalizowane były wzdłuż 239 ulic (podzielonych na 622 segmenty obliczeniowe) i przyłączone do 73 szaf zasilająco-sterujących.

W rezultacie prac badawczo wdrożeniowych związanych z projektem *Pilotażowa modernizacja oświetlenia...* (zob. punkt (d) niniejszego autoreferatu) wykazano, że struktura redukcji zużycia energii osiągniętej dzięki wykorzystaniu zbudowanego przez habilitanta modelu jest następująca.

Lp.	Metoda	Redukcja mocy
1	Przejsie od oświetlenia wyładowczego, sodowego do źródeł LED	40%
2a	Wielkoskalowa optymalizacja instalacji LED (zał. jednorodności sytuacji oświetleniowych)	7%
2b	Wielkoskalowa optymalizacja instalacji LED (na danych GIS)	15%
3	Wprowadzenie adaptacyjnego modelu sterowania opartego o koncepcję profili oświetleniowych	15%
	RAZEM	62% (wariant 2a), 70% (wariant 2b)

Zagadnienia streszczone w niniejszym autoreferacie podsumowuje **monografia** [A10], która podaje także nowe przykłady zastosowań wprowadzonego formalizmu, jak na przykład problem automatycznej generacji reprezentacji grafowych dla sceny 3D i dla systemu sterowania oświetleniem tej sceny.

W opracowaniu [A10] habilitant przedstawił szczegółowo matematyczne podstawy tego formalizmu, również z uwzględnieniem kwestii złożoności obliczeniowej, która decyduje o praktycznej stosowalności modelu. Dla reprezentacji replikowanych grafów komplementarnych (RCG) i grafów ciętych (*slashed graphs*) opisano przejście od postaci scentralizowanej do rozproszonej i przejście w kierunku odwrotnym. W tym kontekście poświęcono również uwagę takim kwestiom, jak równomierność podziału czy podanie efektywnych postaci algorytmów podziału i scalania, pozwalających nie tylko na teoretyczne potwierdzenie wykonalności operacji, ale również wskazujących praktyczny sposób ich realizacji.

Ważne miejsce monografii zajmuje kompleksowe ujęcie problematyki modelowania obiektów fizycznych. Jak zostało to zasygnalizowane w niniejszym autoreferacie, istnienie spójnego opisu formalnego „sceny” (która może obejmować obszar

całego miasta) jest konieczne dla przygotowania modelu obliczeniowego, zrozumiałego dla systemów komputerowych. Hierarchiczny formalizm opisany w monografii wykazuje swoją przydatność między innymi dzięki możliwości przechodzenia od opisu wysokopoziomowego (zagregowanego) do niskopoziomowego (szczegółowego) na wybranych fragmentach reprezentacji sceny.

W pracy zamieszczono przykład zastosowania proponowanego modelu obliczeniowego do problemu projektowania (optymalizacji) wielkoskalowej instalacji oświetleniowej. Przedstawiony schemat zawiera także charakterystykę systemu agentowego wykonującego rozproszone obliczenia projektowe.

Potwierdzenie efektywności prezentowanego środowiska obliczeniowego stanowi jego zastosowanie do rzeczywistych projektów, których lista przedstawiona jest dalej.

(d) Wykorzystanie wyników prac w praktyce

Według raportu sporządzonego przez firmę Northeast Group, roczny koszt energii elektrycznej zużywanej na oświetlenie zewnętrzne osiągnie do roku 2025 wartość 23,9 – 42,5 mld USD, przy czym globalna liczba punktów świetlnych zbliży się do 340 milionów [C5]. Liczby te pokazują, iż nawet niewielka poprawa jednostkowej efektywności energetycznej może przynieść duży zysk z uwagi na efekt skali. Kilkuletnie prace eksperymentalne w tym zakresie, prowadzone przez habilitanta w ramach grantów doprowadziły do opracowania nowego algorytmu projektowania oświetlenia, który pozwala na ok. 7% redukcję pobieranej mocy (obliczenia prowadzone dla instalacji oświetleniowych w miejscowości Geel w Belgii, we współpracy z belgijskim operatorem sieci energetycznych EANDIS). Założeniem tego podejścia była jednorodność obszaru oświetlanego (np. stała szerokość jezdni) oraz instalacji oświetleniowej (identyczne, równo oddalone od siebie lampy). Takie założenie stosowane jest w dostępnych na rynku programach wspomagających projektowanie, np. DIALux, Ulysse, Agi32 itp. Znaczącym wkładem habilitanta było zapewnienie skalowalności tych obliczeń, której nie posiada wspomniane oprogramowanie: obliczenia mogły być wykonywane nie tylko dla pojedynczej ulicy, ale dla obszaru całego miasta. Obliczenia fotometryczne bazujące na grafowych modelach zbudowanych przez habilitanta zostały wykorzystane w następujących projektach.

1. Modernizacja i optymalizacja energetyczna drogowej instalacji oświetleniowej: prace realizowane w ramach międzynarodowego projektu R&D, *Products and Services of a Living Smart Energy City Lab*, dla instalacji oświetleniowych na terenie Flandrii (Belgia).
2. Kierowany przez habilitanta pilotażowy projekt systemu inteligentnego oświetlenia (oparty o system LightGrid) na terenie kampusu AGH, realizowany we współpracy i współfinansowany przez GE Lighting.
3. Projektowanie wielkoskalowych „inteligentnych” instalacji oświetleniowych i systemów ich sterowania:
 - (a) **Projekt SOWA** (współfinansowany ze środków NFOŚiGW) – wymiana ok. 5500 punktów świetlnych na terenie m. Krakowa, poprzedzona optymalizacją konfiguracji instalacji oświetleniowych.
 - (b) **Projekt Pilotażowa modernizacja oświetlenia ulicznego Miasta Krakowa wraz z rozbudową warstwy telemetrycznej, stworzeniem systemu sterowania oraz budową instalacji PV** (współfinansowany ze środków NFOŚiGW oraz Gminę Miejską Kraków) – wymiana ok. 4000 punktów świetlnych na terenie m. Krakowa oraz wdrożenie inteligentnego systemu sterowania, we współpracy z NFOŚiGW ZIKiT i Schreder Polska, poprzedzona optymalizacją konfiguracji instalacji oświetleniowych i projektowaniem systemu sterowania opartego o profile oświetleniowe.
 - (c) **Projekt LIRE** – wymiana (tzw. retrofit) i inteligentne sterowanie ok. 1800 punktów świetlnych na terenie gminy Zielonki, współpraca z GE Lighting, Tauron Dystrybucja.
 - (d) Projekt wymiany 700 opraw sodowych na oprawy LED w Pabianicach (woj. łódzkie). Efekt energetyczny inwestycji objęty jest klauzulą poufności.
4. W trakcie praktycznej weryfikacji znajdują się dwie koncepcje oparte o wyniki otrzymane przez habilitanta:
 - (a) Systemy (typu CAD, BIM - Building Information Modeling) projektowania architektonicznego [A9,B10], w których wykorzystuje się stworzoną przez habilitanta hierarchiczną hipergrafową reprezentację przestrzeni (HHRP), dla modelowania zewnętrznych cech projektowanych obiektów, oraz będącą także wynikiem badań habilitanta, reprezentację grafów ciętych, dla zapewnienia wydajności transformacji grafowych.
 - (b) Automatyczna generacja tzw. grafów sterowania, w której korzysta się z HHRP.

5. Omówienie pozostałych osiągnięć naukowo – badawczych: udział w projektach

1. **Od 2015** – Międzynarodowy projekt *Predictable & Flexible Molten Salts Solar Power Plant* (PreFlexMS), realizowany w ramach perspektywy Horizon 2020.
2. **Od 2015** – Wykonawca merytoryczny w projekcie *Pilotażowa modernizacja oświetlenia ulicznego Miasta Krakowa wraz z rozbudową warstwy telemetrycznej, stworzeniem systemu sterowania oraz budowa instalacji PV*, realizowanym we współpracy z ZIKiT w Krakowie.
3. **Od 2015** – Międzynarodowy projekt współfinansowany przez KIC InnoEnergy, *Pilot implementation of an energy-efficient lighting in a rural or suburban environments*, realizowany przez konsorcjum AGH, Tauron, General Electric, Gmina Zielonki.
4. **Od 2014** – Międzynarodowy projekt współfinansowany przez KIC InnoEnergy (7.7.120.7050), *Long life interconnected smart battery system for off-grid applications* (FOGA).
5. **Od 2013** – Grant Narodowego Centrum Badań i Rozwoju (17.17.120.87010) *Innowacyjna elektrownia wiatrowa z pionową osią obrotu, o mocy 1,5 MW wyposażona w układ przekształtnikowy umożliwiający uzyskanie wysokiej efektywności przetwarzania energii wiatru w energię elektryczną przy niskich prędkościach wiatru*.
6. **2013-2014** – Kierownictwo mini-grantów w ramach projektu SPIN - Model transferu innowacji w Małopolsce (502.120.2066):
 - Wykonanie formalnego modelu danych dla obliczeń fotometrycznych i jego analiza.
 - Opracowanie aspektów techniczno - ekonomicznych wdrażania inteligentnych systemów oświetleniowych.
 - Przeprowadzenie analizy efektywności zastosowania zjawiska „ambient lighting” w inteligentnych systemach oświetleniowych - opracowanie modelu matematycznego na bazie europejskich norm oświetleniowych, oraz przeprowadzenie testów symulacyjnych.
7. **2012** – Przygotowanie WP6 zwycięskiego projektu w ramach FP7 (*Novel technologies and management solutions for protection of crowds*), dotyczącego problemów bezpieczeństwa: Simulation tool for event configuration, intervention strategies and evacuation plans. Projekt koordynowany przez firmy Verint oraz Ernst & Young.
8. **Od 2012** – Kierownictwo zespołu ds. oświetlenia w projekcie strategicznym *Zintegrowane Laboratorium Projektowo-Operacyjne Inteligentnych Infrastruktur Energetycznych - Green AGH Campus*, przygotowywanym we współpracy m.in. z firmą General Electric, Małopolskim Urzędem Marszałkowskim oraz innymi partnerami przemysłowymi i instytucjonalnymi. Aktualnie wystartował kierowany przez habilitanta pilotażowy projekt systemu inteligentnego oświetlenia (oparty o system LightGrid) na terenie kampusu AGH, realizowany we współpracy z GE Lighting. Wybrane publikacje:
 - (a) T. Szmuc, L. Kotulski, B. Wojszczyk, and A. Sędziwy, Green AGH Campus, w: Proceedings of SMARTGREENS 2012, ss. 159–162, 2012,
 - (b) A. Sędziwy, L. Kotulski, *Supporting Fault Tolerance in Graph-Based Multi-Agent Computations*, Lecture Notes in Computer Science, vol. 7895, ss. 397–406, Springer, 2013.
9. **2012-2014** – Projekt Narodowego Centrum Badań i Rozwoju: *Nowoczesne technologie dla/w procesie karnym i ich wykorzystanie – aspekty techniczne, kryminalistyczne, kryminologiczne i prawne*. Publikacja: L. Kotulski, S. Ernst, R. Klimek, R. Mrówka, A. Sędziwy, I. Wojnicki. *Problemy bezpieczeństwa w inteligentnych sieciach energetycznych*, w: Nowoczesne systemy łączności i transmisji danych na rzecz bezpieczeństwa. Szanse i zagrożenia, pod red.: Andrzej R. Pach, Zbigniew Rau, Michał Wągrowski, Wyd. Wolters Kluwer SA, 2013
10. **2011-2013** – Aktywne uczestnictwo w przygotowaniu wniosku o grant w konkursie organizowanym przez KIC InnoEnergy, zakończone otrzymaniem grantu na realizację projektu *Products and Services of a Living Smart Energy City Lab – Alive & KIC-ing*, a następnie pełnienie funkcji koordynatora tematu projektowania oświetlenia w ramach WP6 tegoż projektu, realizowanego przez konsorcjum instytucji i firm z państw UE, współfinansowanego przez węzeł wiedzy i innowacji KIC InnoEnergy. Wybrane publikacje:

- (a) A. Sędziwy, *Effective Graph Representation for Agent-Based Distributed Computing*, Lecture Notes in Computer Science, vol. 7327, ss. 638–647, Springer, 2012,
 - (b) A. Sędziwy, *Representation of Objects in Agent-Based Lighting Design Problem*, Advances in Intelligent and Soft Computing, vol.170, ss. 209–223, Springer, 2012,
 - (c) A. Sędziwy, M. Kozień-Woźniak, *Computational Support for Optimizing Street Lighting Design*, Advances in Intelligent and Soft Computing, vol.170, ss. 241–255, Springer, 2012,
 - (d) L. Kotulski, J. De Landtsheer, S. Penninck, A. Sędziwy, I. Wojnicki, *Supporting energy efficiency optimization in lighting design process*, Proc. of 12th European Lighting Conference, LUX EUROPA 2013
11. **2009-2011** – Grant Narodowego Centrum Nauki: *Formalna metodyka realizacji systemów czasu rzeczywistego z wykorzystaniem wizualnej obiektowej algebry procesów*, realizowanym pod kierunkiem dr. hab. Marcina Szpyrki. Wybrane publikacje:
- (a) L. Kotulski, M. Szpyrka, A. Sędziwy, *Labelled transition system generation from Alvis language*, Lecture Notes in Artificial Intelligence, vol. 6881, ss. 180–189, Springer, 2011,
 - (b) A. Sędziwy, L. Kotulski, M. Szpyrka, *Formal Methods Supporting Agent Aided Smart Lighting Design*, Advances in Intelligent and Soft Computing, vol. 170, ss. 225–239, Springer, 2012,
 - (c) L. Kotulski, M. Szpyrka, A. Sędziwy, K. Grobler-Dębska, *On generation of composite labelled transition systems for alvis passive agents*, The Scientific and Technical Journal „Radioelectronic and Computer Systems”, 2012 [<http://goo.gl/ATYyAa>]
12. **1995-1997** – Projekt Komitetu Badań Naukowych 2P03B08308: *Hadronizacja partonów i ewolucja plazmy kwarkowo-gluonowej* – numeryczne rozwiązywanie nieliniowych cząstkowych równań różniczkowych nieliniowego modelu σ , opisujących ewolucję plazmy kwarkowo-gluonowej.

6. Podsumowanie

Zainteresowania naukowe habilitanta skupione są wokół wykorzystania rozproszonych reprezentacji i transformacji grafowych dla rozwiązywania problemów wysokiej złożoności. Początkowo, prowadzone prace miały charakter teoretyczny, a obszar ich potencjalnego zastosowania ograniczał się do wspomagania systemów typu CAD [A9]. W okresie tym habilitant napisał szereg prac kładących podstawy pod formalizm tzw. grafów komplementarnych [A1]. W szczególności wykazał wielomianową (kwadratową) złożoność jego podstawowych algorytmów, decydującą o możliwości praktycznego zastosowania modelu. Na kolejnym etapie prac, równoległe do indywidualnie stworzonej koncepcji grafów ciętych (*slashed graphs*) [A2], w obszarze zainteresowań habilitanta pojawiła się kwestia projektowania energooszczędnych systemów oświetleniowych [A4,A8]. Zaproponował on nowatorskie podejście do obliczeń fotometrycznych [A5] dające 15% spadek poboru mocy przy jednoczesnym spełnieniu norm oświetleniowych. Mimo doskonałych własności nowej metody, pozostawałaby ona bezużyteczna dla praktycznych zastosowań z uwagi na swoją złożoność obliczeniową. Zastosowanie formalizmu grafów ciętych pozwalające na redukcję złożoności obliczeniowej do poziomu umożliwiającego praktyczne zastosowanie nowego sposobu projektowania, także w zakresie sterowania [A6,A7,B11] jest przedmiotem złożonego wniosku grantowego (status wniosku: w oczekiwaniu na rozstrzygnięcie konkursu).

Niezależnym kierunkiem badań prowadzonych przez habilitanta była budowa uniwersalnego formalizmu hipergrafowego, pozwalającego na modelowanie obiektów przestrzeni 3D [A3]. Stanowiło to odpowiedź na zapotrzebowanie, jakie ujawniły problemy projektowania systemów oświetlenia zewnętrznego, ich optymalizacji energetycznej i sterowania.

Należy nadmienić, że wyniki tych badań znalazły praktyczne zastosowanie w ramach kilku projektów o charakterze wdrożeniowym, m.in.: (i) *Products and Services of a Living Smart Energy City Lab* (optymalizacja energetyczna oświetlenia zewn. na terenie Geel, Belgia), (ii) *Pilot implementation of an energy-efficient lighting in a rural or suburban environments* (projektowanie wymiany ok. 1800 punktów świetlnych i sterowania na terenie gminy Zielonki) czy (iii) *Pilotażowa modernizacja oświetlenia ulicznego Miasta Krakowa wraz z rozbudową warstwy telemetrycznej, stworzeniem systemu sterowania oraz budowa instalacji PV* (projektowanie wymiany ok. 4000 punktów świetlnych i sterowania na terenie m. Krakowa)

Oba wymienione kierunki badań zostały połączone w jeden pełny i spójny model obliczeniowy, który habilitant przedstawił w swojej monografii [A10].

Bibliografia

Publikacje, których autorem lub współautorem jest habilitant, do jakich odwołuje się niniejszy autoreferat, nie wchodzące do wykazywanego osiągnięcia

- [B1] L. Kotulski, A. Sędziwy, *Parallel Graph Transformations Supported by Replicated Complementary Graphs*, Lecture Notes in Computer Science, vol. 6594, ss. 254–264, Springer, 2011
- [B2] L. Kotulski, A. Sędziwy, *Inter-agent communication protocol supporting distributed graph transformations*, w: *Problems of Dependability And Modelling*, (Monographs of System Dependability; vol. 4); eds. Jacek Mazurkiewicz et al.; Oficyna Wydawnicza Politechniki Wrocławskiej, 2011
- [B3] L. Kotulski, A. Sędziwy, B. Strug, *Problem of Agents Cooperation in Heterogeneous Graph-Based Knowledge Environment*, Lecture Notes in Computer Science, vol. 8468, ss. 269–277, Springer, 2014
- [B4] L. Kotulski, A. Sędziwy, *On the Effective Distribution and Maintenance of Knowledge Represented by Complementary Graphs*, Transactions on Computational Collective Intelligence VI, Lecture Notes in Computer Science, vol. 7190, ss. 105–120, Springer, 2012
- [B5] A. Sędziwy, *Effective Graph Representation for Agent-Based Distributed Computing*, Agent and Multi-Agent Systems. Technologies and Applications, Lecture Notes in Computer Science, vol. 7327, ss. 638–647, Springer, 2012
- [B6] A. Sędziwy, L. Kotulski, *Graph-Based Optimization of Energy Efficiency of Street Lighting*, Lecture Notes in Computer Science, vol. 9120, ss. 515–526, Springer, 2015
- [B7] A. Sędziwy, *On acceleration of multi-agent system performance in large scale photometric computations*, Frontiers in Artificial Intelligence and Applications, vol. 252: Advanced Methods and Technologies for Agent and Multi-Agent Systems, ss. 58 – 67, IOS Press, 2013
- [B8] A. Sędziwy, M. Kozień-Woźniak, *Computational Support for Optimizing Street Lighting Design*, Advances in Intelligent and Soft Computing, vol. 170, ss. 241–255, Springer, 2012
- [B9] L. Kotulski, A. Sędziwy, B. Strug, *Translation of Graph-based Knowledge Representation in Multi-agent System*, Procedia Computer Science, vol. 29, ss. 1048–1056, Elsevier, 2014
- [B10] L. Kotulski, A. Sędziwy, B. Strug, *Synchronization mechanism for graph grammar based design*, Proc. of the 20th Int'l Workshop: Intelligent Computing in Engineering, 2013, Vienna University of Technology, ss. 1–10, München: European Group for Intelligent Computing in Engineering, 2013
- [B11] I. Wojnicki, L. Kotulski, S. Ernst, A. Sędziwy, B. Strug. *A Two-Level Agent Environment for Intelligent Lighting Control*, International Journal of Materials and Product Technology, Vol. 53, 2016, Inderscience Enterprises Ltd. (IF: 0.261, 15 pkt.)

Publikacje innych autorów

- [C1] M. Flasiński, *On the parsing of deterministic graph languages for syntactic pattern recognition*, Pattern Recognition, Volume 26, Issue 1, January 1993, Pages 1-16, ISSN 0031-3203, [http://dx.doi.org/10.1016/0031-3203\(93\)90083-9](http://dx.doi.org/10.1016/0031-3203(93)90083-9)
- [C2] D. Janssens, G. Rozenberg, *On the structure of node-label-controlled graph languages*, Information Sciences, Volume 20, Issue 3, April 1980, Pages 191-216, ISSN 0020-0255, [http://dx.doi.org/10.1016/0020-0255\(80\)90038-9](http://dx.doi.org/10.1016/0020-0255(80)90038-9)
- [C3] L. De Floriani, B. Falcidieno, *A hierarchical boundary model for solid object representation*, ACM Transactions on Graphics, vol. 7, ss. 42-60, 1988, <http://dx.doi.org/10.1145/42188.46164>
- [C4] A. Hunt, D. Thomas, *The Pragmatic Programmer: From Journeyman to Master*, Addison-Wesley Longman Publishing Co., Inc., 2000, Boston, MA, USA
- [C5] Echelon, *Making Outdoor Lighting More Efficient, Safe, and Affordable With Open-Standard Control Networking Technology* – Whitepaper, 2015 <<http://goo.gl/mfpGYT>> [Ostatni dostęp: 14 marca 2016 r.]