

POLITECHNIKA POZNAŃSKA

Wydział Informatyki i Telekomunikacji

Instytut Informatyki



Rozprawa doktorska

**METODY PORÓWNYWANIA MODELI
ZŁOŻONYCH SYSTEMÓW BIOLOGICZNYCH
WYRAŻONYCH ZA POMOCĄ SIECI PETRIEGO**

Bartłomiej Szawulak

Promotor: prof. dr hab. inż. Piotr Formanowicz

Poznań 2024

Podziękowania

Serdecznie dziękuję mojemu promotorowi, prof. dr hab. inż. Piotrowi Formanowiczowi,
za powierzenie ciekawego tematu badań, wspólne dyskusje, wyrozumiałość i
za poświęcony czas w trakcie tej wieloletniej współpracy.

Następnie kieruję podziękowania do swoich przyjaciół za ich wsparcie udzielane słowem i
czynem na różnych etapach tej naukowej podróży.

Szczególnie dziękuję mojej partnerce Ali za cierpliwość i wsparcie,
dzięki którym mogłem spełniać swoje marzenie.

Bartłomiej Szawulak

Streszczenie

Biologia systemów reprezentuje szybko rozwijającą się dziedzinę nauki, korzystającą z jednoczesnego rozwoju biologii i informatyki. Obserwowany jest stały wzrost liczby modeli przedstawiających skomplikowane procesy biologiczne. Do przeprowadzenia ich analizy konieczne są dedykowane narzędzia i algorytmy. W zależności od formy reprezentacji matematycznej użytej dla danego modelu, dostępny będzie inny zbiór dostępnych metod analitycznych. W części przypadków możliwa jest transformacja modelu do innej postaci w celu przeprowadzenia dostępnej dla niej analizy. Niestety często jest to obciążone pewnym kosztem precyzji wyniku lub czasochłonności analizy. Powoduje to zapotrzebowanie na dedykowane metody dla danej reprezentacji matematycznej.

Problemem badawczym niniejszej rozprawy jest opracowanie metod służących do ustalania stopnia podobieństwa pomiędzy modelami systemów biologicznych wyrażonych za pomocą sieci Petriego. Sieci Petriego służą do modelowania równoległych procesów dyskretnych, które mogą rywalizować o dostęp do zasobów. W swojej klasycznej formie są one przedstawiane jako dwudzielny, skierowany graf ważony, w którym występują dwa rodzaje wierzchołków. W biologii systemowej sieci Petriego znajdują zastosowanie w modelowaniu sieci metabolicznych, ścieżek sygnałowych oraz innych złożonych systemów biologicznych. W dotychczasowych podejściach analizy modeli przedstawionych za pomocą sieci Petriego widoczny jest brak metod dokonujących oceny podobieństwa dwóch modeli. Zostało to zaobserwowane nie tylko dla systemów biologicznych, ale i w innych obszarach zastosowań sieci Petriego. Do momentu rozpoczęcia badań będących podstawą niniejszej rozprawy, opublikowana była tylko jedna metoda porównywania przepływów w sieciach metabolicznych wyrażonych za pomocą sieci Petriego. Wraz z rosnącą liczbą dostępnych modeli i technik transformacji grafów do postaci sieci Petriego, potrzeba porównywania tych modeli staje się coraz bardziej istotna. Dostęp do odpowiednich metod analizy był dotąd ograniczony.

Rozprawa koncentruje się na opracowaniu metod porównywania klasycznych sieci Petriego dedykowanych do modeli systemów biologicznych. W tym celu wykorzystano obszerną literaturę naukową zajmującą się problemem porównywania grafów i przeanalizowano dostępne w niej metody pod względem ich adaptacji do sieci Petriego, jak i zastosowania w kontekście biologicznym. Metody porównywania sieci Petriego przedstawione w ramach tej rozprawy przedstawiają zróżnicowane podejścia do oceny podobieństwa. Z tego powodu są one dedykowane dla sieci o różnych rozmiarach, strukturze, rozmiarze zbioru niezmienników, oraz kontekstu biologicznego. Metody oparte o zdefiniowane w ramach tej rozprawy wierzchołki rozgałęziające, oraz dedykowany do sieci Petriego wariant grafletów reprezentują podejścia stworzone do działania na dużych sieciach zawierających kilka tysięcy wierzchołków poprzez analizę rozkładu małych podsieci w nich występujących. Reprezentują one podejścia za pomocą których nie jest możliwe określenie izomorfizmu (identyczności strukturalnej), a jedynie ich przybliżonego podobieństwa. Podejściem, które pozwala na określenie izomorfizmu pomiędzy po-

równywanymi sieciami Petriego jest metoda bazująca na dekompozycji do podsieci wybranego typu, dla których wyznaczane są wspólne podsieci. Opracowano kilka algorytmów pozwalających na zastosowanie metody do różnych typów podsieci używanych w modelach biologicznych, jak np. podsieci indukowane przez zbiory MCT i conADT. Metoda ta przez parametryzację pozwala na modyfikację uzyskanej wspólnej podsieci do postaci pożądanej w danej aplikacji biologicznej. Przeprowadzono także modyfikacje metody porównywania modeli ścieżek metabolicznych wyrażonych za pomocą sieci Petriego przez dopasowanie występujących w nich t-niezmienników odpowiadających przepływowi. Dostosowaną ją do działania na t-komponentach, p-niezmiennikach i sub-, sur-niezmiennikach. Wyniki porównano ze zwracanymi przez oryginalny algorytm.

Dla modeli systemów biologicznych wyrażonych za pomocą sieci Petriego nie opracowano powszechnie uznanych norm reprezentacji modelowanych procesów. Z tego powodu szczególną uwagę poświęcono problemowi różnic mających swoje źródła w poziomie szczegółowości i stylach reprezentacji autorów porównywanych modeli.

Niniejsza praca stanowi wprowadzenie do dalszych badań nad porównywaniem złożonych modeli systemów biologicznych.

Abstract

System biology represents an expanding field of science that benefits from the concurrent development of biology and computer science. There is a continuous increase in the number of models describing complex biological processes. Analyzing these models requires specialized tools and algorithms. Depending on the mathematical representation used for a given model, different analytical methods are available. In some cases, it is possible to transform the model into a different type to conduct available analyses, though this often comes with a downgrade of precision or increase of time complexity. This creates a demand for dedicated methods for specific mathematical representations.

The research problem addressed in this dissertation is the development of methods for determining the degree of similarity between biological system models represented using Petri nets. Petri nets are used to model concurrent discrete processes that may compete for access to resources. In their classical form, they are represented as a bipartite, directed weighted graph with two types of vertices.

In systems biology, Petri nets are applied to model metabolic networks, signaling pathways, and other complex biological systems. Previous approaches to analyzing Petri net models have lacked methods for assessing the similarity between two models. This gap has been observed not only in biological systems but also in other areas where Petri nets are applied. Before the start of the research that is a foundation of this dissertation, only one method for comparing flows in metabolic networks represented using Petri nets had been published. Given the continuously growing number of available models and the techniques for transforming graphs of various types into Petri net form, the ability to compare them is a desired analytical method that has previously been limited.

This dissertation focuses on developing methods for comparing classical Petri nets dedicated to biological system models. To achieve this, extensive scientific literature on graph comparison was reviewed, and available methods were analyzed for their adaptation to Petri nets and their application in a biological context. The Petri net comparison methods presented in this dissertation offer diverse approaches to assessing similarity. They are designed for networks of different sizes, structures, minimal invariant set sizes, and biological contexts. Methods based on branching vertices defined in this dissertation, as well as a Petri net variant of graphlets, represent approaches created to work with large networks containing several thousand vertices by analyzing the distribution of small subnets within them. These methods provide approximate similarity and can't confirm an isomorphism. An approach that allows for determining isomorphism between compared Petri nets is based on subnet decomposition of a chosen type, for which common subnets are identified. Several algorithms have been developed to apply this method to various types of subnets used in biological models, such as subnets induced by MCT and conADT sets. This method allows for parameterization to modify the resulting common subnet to the desired form for a given biological application. Additionally, modifications have been made to the method for comparing metabolic pathway models represented using

Petri nets by matching their t-invariants that correspond to metabolic flows. It has been adapted to work with t-components, p-invariants, and sub- and sur-invariants, and the results were compared with those from the original algorithm.

For biological system models represented using Petri nets, there are no widely accepted norms for the representation of modeled processes. Because of that, particular attention has been given to issues arising from differences in the level of detail and the representation styles of the authors of compared models.

This work serves as an introduction to further research on comparing complex biological system models.

Spis treści

1	Wstęp	1
1.1	Wprowadzenie	1
1.2	Cel i zakres rozprawy	3
2	Podstawy teorii grafów i sieci Petriego	8
2.1	Grafy	8
2.1.1	Podstawowa terminologia	8
2.1.2	Rodzaje grafów	10
2.2	Sieci Petriego	12
2.2.1	Historia powstania i rozwój sieci	12
2.2.2	Teoria	15
2.2.3	Redukcje sieci	18
3	Przegląd metod porównywania grafów i sieci	20
3.1	Izomorfizm	20
3.1.1	Teoria	20
3.1.2	Historia	21
3.2	Metody porównywania grafów	21
3.2.1	Dystans grafowy	22
3.2.2	Największy wspólny podgraf	23
3.2.3	Grafowa odległość edycyjna	23
3.3	Grafilety	24
3.3.1	Historia	24
3.3.2	Digrafilety	25
3.3.3	Metryki grafiletowe	25
3.4	Porównywanie przez niezmienniki	28
3.4.1	Historia rozwoju	28
3.4.2	Obszary zastosowań	29
3.4.3	Metoda składowa bazująca na wartościach EC	29
3.4.4	Metoda składowa bazująca na niezmiennikach	29
3.4.5	Algorytm	30

4	Dekompozycja sieci Petriego	31
4.1	Wstęp	31
4.2	Dekompozycja sieci Petriego	31
4.2.1	t-komponent i s-komponent	32
4.2.2	Zbiory MCT	33
4.2.3	Zbiory ADT	34
4.2.4	t-sieć	35
4.2.5	Zbiory MCP	36
4.2.6	Spójne zbiory ADP	36
4.2.7	s-sieć	36
4.2.8	P1-sieć / komponent maszyny stanowej	37
4.2.9	Podsieci funkcyjne	38
4.3	Dekompozycje bazujące na fragmentach t-komponentów	38
4.3.1	Rodzina MCT, conADT i t-sieć	39
4.4	Dekompozycje bazujące na fragmentach p-niezmienników	40
4.4.1	Rodzina MCP, conADP i s-sieć	40
4.4.2	P1-sieć	41
4.5	Podsieci funkcyjne	42
5	Wierzchołki rozgałęziające	44
5.1	Definicje	44
5.2	Wierzchołki rozgałęziające a algorytmy grafowe	47
5.2.1	Wierzchołki rozgałęziające a niezmienniki w sieciach Petriego	47
5.2.2	Wierzchołki rozgałęziające a podsieci ADT/MCT/t-sieci	48
5.2.3	Wierzchołki rozgałęziające a podsieci funkcyjne	50
6	Grafloty w sieciach Petriego	52
6.1	Wprowadzenie	52
6.2	Modyfikacje struktur graflotów	52
6.3	Wyznaczanie rozkładu	56
6.4	RGF	56
6.5	GDDA	60
6.5.1	Wyznaczanie GDDA	61
6.5.2	Zachowanie GDDA w sieciach Petriego	62
6.5.3	Stabilność GDDA	74
7	Porównanie z użyciem niezmienników	78
7.1	Wprowadzenie	78
7.1.1	Przyczyny wprowadzonych modyfikacji	78
7.1.2	Modyfikacje	78
7.2	Przykłady	80
7.2.1	Przykład I – porównanie jako rozszerzenie analizy wyłączania podsieci	80
7.2.2	Przykład II – sub-/sur- t-niezmienniki i ich wpływ na wynik porównania	82

7.2.3	Przykład III – mapowanie etykiet i niezmienników	84
7.3	Podsumowanie	87
8	Porównanie z użyciem wierzchołków rozgałęziających	90
8.1	Wprowadzenie	90
8.2	Pomysł i inspiracja	90
8.3	Porównanie z grafletami	91
8.4	Algorytm	94
8.5	Przykłady	95
8.5.1	Małe sieci metaboliczne	95
8.5.2	Modele złożonych systemów biologicznych	98
8.6	Podsumowanie	102
9	Porównanie przez dekompozycje	103
9.1	Wprowadzenie	103
9.1.1	Krótką historia rozwoju	103
9.1.2	Idea	103
9.2	Niematematyczne wyzwania	104
9.2.1	Redukcja sieci	105
9.2.2	Problem największego wspólnego podgrafu – biologiczna perspektywa	105
9.2.3	Wspólna podsieć	106
9.3	Proces porównywania podsieci	108
9.3.1	Format wyniku	108
9.3.2	Algorytm	109
9.3.3	Dostępne warianty wspólnych podsieci	112
9.4	Przykład zastosowania	115
9.4.1	Przypadki trywialne	118
9.4.2	Częściowe dopasowanie	119
9.4.3	Pełne dopasowanie	121
9.4.4	Porównanie sieci po redukcjach strukturalnych	122
9.5	Zastosowanie do pokrewnych dekompozycji opartych o niezmienniki	126
9.5.1	conADT	127
9.5.2	MCT	127
9.5.3	T/S-komponent	127
9.5.4	ADP/MCP – podsieci bazujące na p-niezmiennikach	127
9.5.5	s-sieci	128
9.6	Porównanie przez dekompozycję – podsieci funkcyjne	128
9.6.1	Podsieć	129
9.6.2	Algorytm	130
9.6.3	Przykład	130
9.7	Podsumowanie	132

10 Wyniki uzupełniających eksperymentów obliczeniowych	133
10.1 Wprowadzenie	133
10.2 Uzupełnienie przypadku – modele naprawy DNA	133
10.2.1 Grafloty	133
10.2.2 Relative Branching Vertices Frequency distance – RBF	134
10.3 Uzupełnienie przypadku – sieci chorobowe	136
10.3.1 Grafloty – GDDA	136
10.3.2 Porównania podsieci t-sieci	138
10.4 Małe sieci metaboliczne	139
10.4.1 Porównywanie z użyciem dekompozycji	139
10.5 Protokoły komunikacji BGP i ECMA	140
10.5.1 Grafloty	140
10.5.2 Dekompozycja do t-sieci i podsieci funkcyjnych	140
10.5.3 Relative Branching Vertices Frequency distance – RBF	142
10.6 Ewaluacja metod	145
10.6.1 Problem różnych źródeł	145
10.6.2 Problem rozmiaru sieci	147
10.6.3 Problem wykładniczego wzrostu liczby cząstkowych wektorów	148
11 Podsumowanie	149
Bibliografia	150

Rozdział 1

Wstęp

1.1 Wprowadzenie

Postępujący w ciągu ostatniego półwiecza rozwój informatyki i towarzysząca mu powszechna informatyzacja w naturalny sposób spowodowały wzrost zainteresowania jej potencjałem pośród naukowców reprezentujących różne dyscypliny naukowe.

Efektem było powstanie nowych dyscyplin na miejscach styku już istniejących, czego przykładami mogą być: chemioinformatyka, astroinformatyka, archeoinformatyka czy będąca głównym tłem tej rozprawy bioinformatyka. Z obszernej dyscypliny, jaką jest bioinformatyka, niniejsza praca skupia się na obszarze współdzielonym z biologią systemową, a dokładnie na porównywaniu modeli złożonych systemów biologicznych.

Bioinformatyka, biochemia, biotechnologia czy biologia molekularna są reprezentantami nauk biologicznych, które łączy kluczowy z perspektywy informatyki element – zapotrzebowanie na matematyczne i algorytmiczne metody pozwalające na znalezienie rozwiązań dla problemów występujących w tych dziedzinach oraz pośrednio moc obliczeniową potrzebną do przeanalizowania rosnących z każdym rokiem wolumenów danych.

Możliwości, jakie daje wykorzystanie informatyki, nie ograniczają się do samego przetwarzania danych uzyskanych z laboratoriów. Zajmuje się ona rozwiązywaniem problemów biologicznych, takich jak sekwencjonowanie genomów czy modelowanie procesów poprzez zastosowanie z perspektywy biologii nowych podejść, będących adaptacją istniejących wcześniej w informatyce metod i algorytmów.

Biologia systemowa to obszar biologii skupiający się na modelowaniu, badaniu i analizowaniu złożonych systemów biologicznych. System biologiczny rozumiany jest jako zbiór komponentów (np. podprocesów o biologicznym znaczeniu, cząsteczek elementarnych, lub reakcji chemicznych) wzajemnie połączonych i oddziaływających na siebie [1]. Na przestrzeni lat wykształcił się szereg metod służących do ich modelowania, takich jak: układy równań różniczkowych zwyczajnych, równań różniczkowych cząstkowych, sieci Petriego, rachunek π , automaty komórkowe, modelowanie agentowe oraz różne modele hybrydowe będące kombinacją kilku z wymienionych podejść [79].

Sieci Petriego są przeznaczone do reprezentacji równoległych dyskretnych procesów, które mogą konkurować między sobą o zasoby. Mają one w klasycznym wariacie reprezentację w postaci dwudzielnego, skierowanego grafu ważonego z dwoma typami wierzchołków. Sieci Petriego są często określane jako intuicyjne narzędzie [30, 56], co przekłada się na ich popularność. Zainteresowanie nimi zaowocowało dużą liczbą narzędzi analitycznych i metod stworzonych przez społeczność naukową [31, 53, 103].

Same sieci Petriego są obecne w wielu różnych obszarach nauki, w których istnieje potrzeba reprezentacji równoległych procesów [6, 12, 84, 91].

W kontekście biologicznym, a konkretnie biologii systemowej, zastosowanie sieci Petriego zachodzi w obszarach sieci metabolicznych, ścieżek sygnałowych czy innych złożonych systemów biologicznych. W przypadku sieci metabolicznych mogą być one zazwyczaj łatwo przetransponowane do postaci sieci Petriego [10, 89, 106]. Na przestrzeni ostatnich trzydziestu lat, od kiedy po raz pierwszy zaproponowano zastosowanie sieci Petriego w biologii, wykorzystane zostały one do zamodelowania szerokiego spektrum procesów biologicznych [24, 25, 35, 37, 52, 69, 89, 92].

Głównym celem niniejszej rozprawy jest przedstawienie opracowanych metod porównywania sieci Petriego. Powodem skupienia się na tym problemie jest istniejące zapotrzebowanie na możliwości porównania modeli systemów biologicznych. Jako przykładowe sytuacje zastosowania mogą posłużyć takie przypadki jak: konieczność ustalenia skali różnic, wraz z ich lokalizacją w porównywanych modelach szlaków metabolicznych; analiza zachowań chorób współistniejących rozważanych jako osobne choroby w celu rozpoznania interakcji pomiędzy nimi oraz układem odporności. Mimo istniejącego zapotrzebowania istniała tylko jedna metoda przeznaczona dla porównywania sieci Petriego [10]. Opiera się ona na porównywaniu poprzez użycie dwóch podmetod, bazujących na porównaniu znalezionych w sieciach niezmienników i dopasowaniu elementów z baz enzymów. Poza obszarem biologicznego zastosowania sieci Petriego, w momencie pisania niniejszej pracy, nie zostały znalezione żadne inne opracowania metod porównywania. Prowadzi to do ciekawego pytania, dlaczego w innych dziedzinach, w których sieci Petriego występują znacznie dłużej, nie powstały żadne dedykowane metody porównywania sieci? Natomiast obszarem, w którym można znaleźć potencjalne rozwiązania rozważanego problemu, jest teoria grafów oraz jej obszary dotyczące izomorfizmu, porównywania lub wyszukiwania struktur.

Teoria grafów ma swoje początki w XVIII wieku, gdy Leonhard Euler wykazał niemożliwość istnienia rozwiązania dla problemu królewieckich mostów [13]. Skupia się ona na badaniu charakterystyk struktur nazywanych grafami. Przedstawiane są one graficznie jako zbiór wierzchołków i łączących je krawędzi. Za ich pomocą możliwe jest przedstawienie różnych skomplikowanych problemów (procesów) w łatwiejszej do interpretacji postaci. Jako przykład może posłużyć model procesu biologicznego który został przedstawiony jednocześnie jako zbiór równań różniczkowych i graficznego modelu sieci Petriego, który nie wymaga znajomości analizy matematycznej. W celu dokładniejszej reprezentacji modelowanego procesu rozważane są różne typy grafów, których charakterystyki wpływają na efektywną reprezentację pożądanых cech.

Problem porównywania sieci Petriego jak i grafów można przedstawić jako proces ustalania stopnia i formy podobieństwa zachodzącego pomiędzy dwoma modelami. Dwa najpopularniejsze podejścia skupiają się na znajdowaniu dystansu (liczby różnic) lub ustaleniu maksymalnej wspólnej struktury. Zastosowaniem praktycznym porównywania, poza oczywistym ustaleniem stopnia podobieństwa, jest na przykład wykrywanie fragmentów sieci odpowiadających niepożądanym podprocesom [95] lub ocena skali i kosztów zmian jakie należy wykonać aby zunifikować dwa modele do jednej postaci [129]. Zaznaczyć należy, że struktury otrzymane w wyniku modelowania procesu biologicznego mogą osiągać znaczące rozmiary (od kilkudziesięciu do kilkuset wierzchołków). Znajdowanie podobieństw pomiędzy nimi nie jest łatwym zadaniem, ponieważ nie są znane wielomianowe algorytmy porównywania dla klas nadrzędnych jak, np. grafy nieskierowane. Jednocześnie istnieją wydajne algorytmy opracowane

dla konkretnych klas grafów, takie jak grafy planarne [124], czy drzewa [23].

Kwestia porównywania grafów jest bardzo ważnym zagadnieniem angażującym wielu naukowców. Częściowo ma to swoje źródło w tym, że problem izomorfizmu grafów pozostaje wciąż otwarty. Według artykułu przeglądowego pod przewodnictwem Emmerta-Streiba [32] pierwszą publikacją proponującą metodę pomiarów różnic pomiędzy dwoma grafami był artykuł [141] Bohdana Zelinki z 1975 roku. Temat ten zaczął następnie zajmować umysły naukowców z ówczesnego bloku wschodniego [62, 117], a później reszty świata. W ciągu pół wieku opracowano i wielokrotnie ulepszano algorytmy dla konkretnych klas grafów, jak i dla ogólnego przypadku. Dzięki temu istnieje obecnie liczny zbiór metod, które mogą zostać zaadaptowane dla nowych typów grafów lub posłużyć jako baza do stworzenia własnych algorytmów.

Dostępne techniki porównywania grafów i sieci reprezentują rodzinę metod o bardzo różnych podejściach do określania poziomu podobieństwa lub różnic, jakie zachodzą pomiędzy dwoma grafami czy sieciami. Te metody różnią się często między sobą pod wieloma względami, za co odpowiedzialny jest kontekst, w którym został stworzony dany algorytm. Wszystkie istniejące metody porównywania można podzielić na dwie klasy w zależności od odpowiedzi na pytanie czy dwa grafy o różnej liczbie wierzchołków mogą być porównane. Pierwsza grupa, pozwalająca na takie porównania nazywana jest niedokładnymi porównaniami (ang. *inexact comparison*), a druga zabraniająca ich, dokładnymi porównaniami (ang. *exact comparison*).

Dokładne dopasowanie jest koniecznym warunkiem dla problemu określenia izomorfizmu grafów. Dwa grafy nazywamy izomorficznymi, jeśli istnieje funkcja przypisująca każdemu wierzchołkowi w pierwszym grafie odpowiadający mu wierzchołek w drugim grafie. Metody dopuszczające niedokładne dopasowania grafów i sieci są znacznie częściej wykorzystywane niż dokładne metody poszukujące izomorfizmu. Dla części z nich kwestia izomorfizmu jako punktu odniesienia jest nadal istotna (np. dystans grafowy i grafowa odległość edycyjna), podczas gdy dla pozostałych nie ma ona znaczenia (np. stopień rozkładu grafletów w sieciach). Większość popularnych metod została dostosowana do działania w przypadkach poszukiwania dokładnego i niedokładnego dopasowania.

Podsumowując, istotne dla społeczności naukowej zajmującej się analizą procesów biologicznych (i nie tylko) za pomocą sieci Petriego jest uzyskanie nowych metod porównywania. Opracowane metody powinny mieć możliwość zastosowania w przypadkach, które dotychczas ze względu na swoją strukturę czy rozmiar nie podlegały procesowi porównywania. Dostarczenie nowych narzędzi, jakimi są opracowane w ramach tej rozprawy algorytmy, ma doprowadzić do wzrostu zainteresowania porównywaniem sieci Petriego.

1.2 Cel i zakres rozprawy

Problemem badawczym niniejszej rozprawy jest analiza skali i formy podobieństwa pomiędzy modelami systemów biologicznych przedstawionymi za pomocą sieci Petriego. W tym celu wykorzystywane są algorytmy powstałe w wyniku adaptacji istniejących rozwiązań z teorii grafów oraz nowe dedykowane rozwiązania.

Pierwszym celem jest lepsze zrozumienie problemu porównywania i oceny skali podobieństwa modeli systemów biologicznych wyrażonych za pomocą klasycznych sieci Petriego. W tym celu przeprowadzone zostały poszukiwania już istniejących w literaturze algorytmów, powtarzane kilkakrotnie w trakcie prowadzenia badań. Rezultatem było potwierdzenie istnienia (wedle wiedzy autora) do momen-

tu zakończenia pisania tej rozprawy tylko jednej metody porównywania sieci Petriego [10]. Wymusiło to kolejny krok, którym było zaczerpnięcie obszernej palety metod porównywania z pokrewnego obszaru nauki jakim jest teoria grafów. Zapoznano się z wieloma rodzajami reprezentacji podobieństwa grafów, jak np. dystans grafowy, grafowa odległość edycyjna czy największy wspólny podgraf, które są obecne w zaproponowanych rozwiązaniach lub mają wpływ na nie. Dystans grafowy jest najstarszą miarą podobieństwa, służącą pierwotnie do potwierdzania zachodzenia izomorfizmu lub wyznaczenia minimalnej liczby różnic (dystansu) w postaci wierzchołków lub łuków, których usunięcie doprowadziłoby do izomorfizmu sieci. Grafowa odległość edycyjna (ang. Graph Edit Distance) bazuje na idei Levenshteina dla znakowej odległości edycyjnej, z tą różnicą, że operuje na grafach zamiast ciągach. Za jej pomocą wyznaczany jest minimalny koszt transformacji jednego grafu w drugi. Największy wspólny podgraf reprezentuje maksymalną strukturę, jaka występuje jednocześnie w dwóch porównywanych grafach. W zależności od algorytmu może on odpowiadać wynikowi ostatecznemu, jak i krokowi pośredniemu. Obserwacją z analizy zastosowań sieci Petriego w różnych obszarach naukowych i przemysłowych są różnice w istotności różnych charakterystyk sieci Petriego w zależności od kontekstu, w którym zostały wykorzystane. Zrodziło to pytanie, czy metody zaczerpnięte z teorii grafów będą wystarczające do oceny skali podobieństwa modeli układów biologicznych. Efektem było zaprezentowanie interpretacji wyników skupiającej się na istotnych z perspektywy biologicznej elementach sieci oraz modyfikacjach algorytmów mających na celu ich identyfikację.

Głównym celem tej rozprawy jest zaprezentowanie rozwiązań dla problemu porównania modeli systemów biologicznych przedstawionych za pomocą sieci Petriego, wraz z analizą problemu i przedstawieniem go w kontekście metod opracowanych dla porównywania grafów. Metody stworzone do oceny poziomu podobieństwa można podzielić na reprezentujące dwa podejścia: adaptacje do nowej sieci Petriego istniejących metod porównywania grafów (ze szczególnym uwzględnieniem metod posiadających historię zastosowań do modeli biologicznych) oraz nowe dedykowane dla sieci Petriego metody. Pierwszym krokiem było znalezienie metod porównywania grafów niebędących w konflikcie z żadną z charakterystyk klasycznych sieci Petriego, np. algorytm stworzony dla sieci trójdzielnych. W efekcie mogą one zostać zmodyfikowane do działania w nowym środowisku. Potencjalne metody były oceniane na bazie dwóch własności, tj. poziomu trudności koniecznych modyfikacji i efektywności algorytmu po ich wprowadzeniu w porównaniu z oryginalnym algorytmem. Powyższe warunki spełniają metody bazujące na grafletach, które odpowiadają małym, nieizomorficznym podgrafom. Metody operujące na nich dokonują oceny podobieństwa na bazie różnic w ich dystrybucjach uzyskanych dla porównywanych sieci. Z dostępnych metryk grafletowych, po przeprowadzeniu szeregu testów, dokonano selekcji wybierając te, których działanie spełniało wymagania dla systemów biologicznych. W krótkiej historii porównywania sieci Petriego pierwszą opracowaną metodą jest heurystyka badająca podobieństwo pomiędzy modelami poprzez dopasowanie występujących w nich niezmienników w pary uznane za reprezentujące podobne przepływy [10]. Metoda ta obarczona jest szeregiem ograniczeń, które w ramach zaproponowanych modyfikacji udało się częściowo zrekompensować. Następnym krokiem jest opracowanie nowych dedykowanych metod porównywania, które w jak największym stopniu wykorzystują własności strukturalne sieci Petriego. Pierwsza metoda bazuje na dekompozycji porównywanych modeli do zbioru rozłącznych podsieci reprezentujących występujące w nich podprocesy. Następnie dla każdej pary dekomponowanych podsieci wyznaczana jest wspólna podsieć. Algorytm dostosowano aby operował na popularnych w modelach biologicznych dekompozycjach jak np. najwięk-

szy wspólny podgraf. Takie podejście przyspiesza analizę modeli poprzez skupienie się na istotnych z biologicznej perspektywy fragmentach sieci. Ze względu na to, że systemy biologiczne odpowiadają bardzo zróżnicowanym procesom, zaimplementowano kilka wariantów algorytmu, pozwalających na dopasowanie poszukiwanej podsieci do wymagań oczekiwanych w danym kontekście biologicznym. Kolejna metoda jest inspirowana podejściem grafletowym. Porównuje ona sieci, badając rozkład wierzchołków rozgałęziających. Wierzchołki rozgałęziające stanowią nowy termin wprowadzony na potrzeby zaproponowanych metod porównywania i są jednocześnie formą standaryzacji części pojęć występujących w różnych obszarach zastosowań sieci Petriego. Zaproponowana metoda uzupełnia i poszerza możliwości analityczne w porównaniu do podejścia grafletowego.

Następnie przeprowadzone jest porównanie przedstawionych algorytmów, ich wydajności i złożoności. Jednocześnie należy pamiętać, że przedstawione metody reprezentują często bardzo różne podejścia do porównywania sieci Petriego. Uzyskane w takich przypadkach rezultaty nie mogą zostać bezpośrednio ze sobą zestawione bez dodatkowych modyfikacji. Dlatego też dla prawidłowej analizy porównawczej metody zostały zebrane w pary lub trójki, w ramach których przeprowadzono właściwe porównania.

Rozprawa składa się z jedenastu rozdziałów:

Rozdział 1 zawiera wprowadzenie do tematyki rozprawy oraz uzasadnienie podjęcia badań i ich znaczenia.

Rozdział 2 zawiera wprowadzenie do podstawowych zagadnień z obszaru teorii grafów. Przedstawia kompleksowo definicję klasycznego wariantu sieci i przybliża postać Carla Adama Petriego wraz z historią rozwoju opracowanych przez niego sieci.

Rozdział 3 zawiera definicje problemu porównywania grafów i izomorfizmu. Przedstawia przegląd istniejących podejść do porównywania grafów, opisując w szczególności z użyciem definicji matematycznych najpowszechniejsze metody takie jak dystans grafowy, największy wspólny podgraf, czy Grafowy Dystans Edycyjny (ang. *Graph Edit Distance* GED). Ponadto, cały podrozdział został poświęcony grafletom. Przedstawiono ich warianty oraz opracowane dla nich miary podobieństwa: Dystans Względnej Częstotliwości Grafletów (ang. *Relative Graphlet Frequency distance* RGF), Rozkład Stopni Grafletów (ang. *Graphlet Degree Distribution* GDD), uzgodniony stopień dystrybucji grafletów (ang. *Graphlet Degree Distribution Agreement* GDDA), NetDis. Rozdział kończy prezentacja metody porównywania sieci Petriego z użyciem niezmienników, będącej do czasu ukończenia tej rozprawy jedyną metodą porównywania dedykowaną dla sieci Petriego.

Rozdział 4 przedstawia istniejące metody dekompozycji sieci Petriego do krawędziowo-rozłącznych podsieci będących podstawą dla algorytmów porównywania przedstawionych w Rozdziale 9. Przedstawione są relacje zachodzące pomiędzy różnymi dekompozycjami bazującymi na niezmiennikach, wraz z wyszczególnieniem różnic występujących pomiędzy nimi. Poświęcono uwagę kwestii pozornej zamenności metod dekompozycji bazujących na t-niezmiennikach z dekompozycjami bazującymi na p-niezmiennikach.

Rozdział 5 zawiera nową terminologię wierzchołków rozgałęziających wprowadzoną na potrzeby tej rozprawy. Poza formalną definicją i przykładami, przedstawione zostały funkcje tych wierzchoł-

ków w odniesieniu do istniejących już struktur, takich jak t-komponenty, zbiory MCT, czy sieci funkcyjne.

Rozdział 6 przedstawia nową metodę porównywania sieci Petriego opartą o koncept grafletów. Zawiera opis adaptacji grafletów (pn-graflety) do sieci Petriego, jak i skalę wprowadzonych zmian wynikających z ograniczeń jakie narzuciła transformacja. Następnie skupia się na przedstawieniu efektywności dwóch wyselekcjonowanych metod porównywania RGF i GDDA, znanych zastosowań do grafów prostych i digrafów, przy zastosowaniu nowego typu grafletów dedykowanych do sieci Petriego. Na licznych przykładach przedstawiono reakcje obu metod na różne rodzaje różnic występujących w modelach sieci biologicznych. W przypadku GDDA przeprowadzono badania ilościowe na wygenerowanych losowych sieciach, spełniających charakterystyki typowe dla modeli systemów biologicznych. Poświęcono też uwagę analizie niestabilności zaobserwowanej dla GDDA operującej na grafach nieskierowanych o niskiej gęstości. W wyniku przeprowadzonego eksperymentu jej obecność została potwierdzona dla pn-grafletów, jednak skala i postać znacząco różni się od pierwotnie raportowanego problemu dla nieskierowanych grafletów.

Rozdział 7 zawiera opis metody porównywania sieci Petriego bazującej na dopasowaniu t-niezmienników, będącej pierwszym podejściem do tego problemu. Następnie proponowane są zmiany w postaci modyfikacji danych wejściowych: poszerzenie zbioru o p-niezmienniki, przejście z t-niezmienników na t-komponenty mające na celu poprawienie jakości wyników zwracanych przez oryginalny algorytm. Efekty zmian przedstawiono na kilku przykładach, odpowiadających trzem różnym sytuacjom, w których pojawiła się potrzeba porównania sieci z użyciem t-niezmienników. Zwrócono uwagę na największą słabość metody, dotyczącą identyfikacji dopasowywanych wierzchołków na podstawie dedykowanej zewnętrznej bazy danych.

Rozdział 8 zawiera opis nowej metody bazującej na rozkładzie wierzchołków rozgałęziających. Stanowi ona wariację RGF opisaną szczegółowo w Rozdziałach 3 i 6 do operowania na strukturach opisywanych przez wierzchołki rozgałęziające. Korzyści ze zmiany struktur wejściowych na nowe są przedstawione w porównaniu z pn-grafletami, pozwalając określić przypadki, w których użycie wierzchołków rozgałęziających daje lepsze rezultaty.

Rozdział 9 zawiera opis nowej metody bazującej na dekompozycji porównywanych modeli do rozłącznych podsieci, dla których wyznaczane są następnie wspólne podsieci. Pod dyskusję poddawana jest kwestia znaczenia biologicznego największych wspólnych podsieci, która konfrontowana jest z oczekiwaniami stawianymi przez autorów modeli biologicznych. Rezultatem jest propozycja wspólnej podsieci maksymalizującej liczbę wierzchołków rozgałęziających. Następnie przedstawiony jest algorytm porównywania sieci i jego modyfikacje pozwalające na uzyskanie kilku przewidzianych wariantów wspólnych podsieci, mających pozwolić uzyskać dokładniejsze wyniki w zależności od kontekstu biologicznego porównywanych modeli i wymagań autorów. Za pomocą przykładowych sieci przedstawiono różne warianty wspólnych podsieci, jakie można wyznaczyć pomiędzy porównywanymi podsieciami. Przedstawiono wpływ redukcji struktury modeli na uzyskiwane wyniki. Opisano modyfikacje konieczne do dostosowania algorytmu do operowania na różnych typach dekomponowanych podsieci. Opracowany został także nowy algorytm dedykowany dla podsieci funkcyjnych, który bazuje na wspólnej idei porównania zdekomponowanych fragmentów sieci. Ze względu na różnice strukturalne pomiędzy podsieciami funkcyjnymi

a rodziną podsieci bazujących na niezmiennikach, konieczne było opracowanie nowego algorytmu porównującego dwie podsieci.

Rozdział 10 zawiera opis i wyniki eksperymentów porównawczych będących uzupełnieniem przypadków z Rozdziałów 6-9. Do tego celu wykorzystano zbiór modeli systemów biologicznych, z przykładów jakie wystąpiły na łamach tej pracy. Jeśli na danej parze modeli w poprzednich rozdziałach nie zostało wykonane porównanie z użyciem jednej z opracowanych metod, to w ramach uzupełnienia przeprowadzono je w tym rozdziale. Na bazie wyników porównań dokonano ewaluacji każdej z metod, uwzględniając jej mocne i słabe strony.

Rozdział 11 zawiera podsumowanie rozprawy, w szczególności ocenę przedstawionych metod oraz ich potencjalne zastosowanie.

Rozdział 2

Podstawy teorii grafów i sieci Petriego

W tym rozdziale znajdują się podstawowe definicje z zakresu teorii grafów i sieci Petriego używane w dalszej części pracy. Są one niezbędne do poprawnej interpretacji metod porównywania przedstawionych w Rozdziałach od 6 do 9.

2.1 Grafy

W niniejszym podrozdziale zostaną przytoczone podstawowe definicje z zakresu teorii grafów, kluczowe dla tematów poruszanych w kolejnych rozdziałach. Z powodu istnienia w literaturze wielu różnych definicji dotyczących grafów konieczne było uniknięcie potencjalnych rozbieżności przez przyjęcie jednej spójnej postaci. Prezentowane w tym podrozdziale podstawowe definicje opisujące grafy są przedstawione w notacji matematycznej bazującej na stosowanej w książce [42].

2.1.1 Podstawowa terminologia

Definicja 1. Graf

Niech V będzie niepustym zbiorem i niech $E \subset \{\{v_i, v_j\} : v_i \in V \wedge v_j \in V\}$. Parę $G = (V, E)$ nazywamy grafem nieskierowanym, przy czym zbiór V nazywany jest zbiorem wierzchołków, a zbiór E nazywany jest zbiorem krawędzi.

Definicja 2. Pętla

Jeśli krawędź e_i posiada jako oba swoje końce wierzchołek v_j , jest nazywana *pętlą własną* $e_i = \{v_j, v_j\} = \{v_j\}$.

Definicja 3. Graf prosty

Graf nazywamy *grafem prostym*, jeśli nie zawiera żadnych pętli i wielokrotnych krawędzi występujących pomiędzy dwoma wierzchołkami.

Definicja 4. Graf skierowany

Niech V będzie niepustym zbiorem i niech $A \subset V \times V$. Parę $G = (V, A)$ nazywamy *grafem skierowanym* lub *digrafem*, przy czym zbiór V nazywany jest zbiorem wierzchołków, a zbiór A nazywany jest zbiorem łuków. Pojedynczy łuk jest uporządkowaną parą wierzchołków $a_i = (v_i, v_j)$, gdzie wierzchołek v_i jest jego początkiem, a v_j końcem.

Definicja 5. Stopień wierzchołka

Dla wierzchołka v_i liczba stycznych z nim krawędzi jest nazywana jego *stopniem* i oznaczana przez $\deg(v_i)$. Odpowiada ona sumie krawędzi zawierających v_i jako jeden ze swoich końców. W przypadku gdy v_i posiada pętlę własną, jest ona liczona podwójnie.

Definicja 6. Sąsiedztwo

Z perspektywy wierzchołka v_i inny wierzchołek v_j , jest nazywany jego *sąsiadem* jeśli oba występują jednocześnie w jednej krawędzi $e = \{v_i, v_j\}$. W grafach prostych stopień wierzchołka v_i odpowiada liczbie jego sąsiadów.

Definicja 7. Macierz sąsiedztwa

Macierz sąsiedztwa A dla grafu G jest macierzą o rozmiarach $n \times n$, gdzie pozycja a_{ij} przechowuje wartość 1, jeśli istnieje krawędź $\{v_i, v_j\}$, a 0 gdy nie istnieje. Na Rysunku 2.1 został przedstawiony przykład grafu i odpowiadającej jej macierzy.



Rysunek 2.1: Przykładowy graf nieskierowany wraz z odpowiadającą mu macierzą sąsiedztwa

Definicja 8. Łańcuch

Łańcuchem $x - y$ w grafie G jest sekwencja wierzchołków i krawędzi mająca końce w wierzchołkach x i y , oraz zawierającą n krawędzi $e_i = \{v_{i-1}, v_i\}$, gdzie $i = 1, 2, \dots, n$. Łańcuch można przedstawić w następującej postaci:

$$x = v_0, e_1, \dots, e_n, v_n = y \quad (2.2)$$

Definicja 9. Droga

Drogą nazywany jest łańcuch, w którym żadna z zawierających się w nim krawędzi nie występuje więcej niż jeden raz.

Definicja 10. Ścieżka

Ścieżką nazywany jest łańcuch, w którym żaden z zawierających się w nim krawędzi i wierzchołków nie występuje więcej niż jeden raz.

Definicja 11. Cykl

Cyklem nazywamy taką ścieżkę, w której wierzchołek początkowy v_0 i końcowy v_n są tym samym wierzchołkiem, tj. $v_0 = v_n$. Graf jest nazywany grafem cyklicznym, jeśli zawiera przynajmniej jeden cykl, a acyklicznym w przypadku gdy nie zawiera żadnego cyklu. Skierowane acykliczne grafy są często skrótowo nazywane *DAG* (ang. Directed acyclic graph)

Definicja 12. Obwód

Obwodem w grafie G nazywamy taką drogę, w której wierzchołek początkowy v_0 i końcowy v_n są tym samym wierzchołkiem, tj. $v_0 = v_n$.

Definicja 13. Graf spójny

Graf G jest nazywany *grafem spójnym*, jeśli dla każdej pary wierzchołków v_i i v_j wchodzących w skład tego grafu istnieje łącząca je ścieżka. Graf niespójny składa się z przynajmniej dwóch spójnych rozłącznych podgrafów, nazywanych *komponentami*. Jeżeli usunięcie pojedynczej krawędzi e tworzy dwa komponenty, taką krawędź nazywamy mostem.

Definicja 14. Graf pełny

Graf nieskierowany, w którym pomiędzy każdą parą wierzchołków istnieje krawędź, nazywamy *grafem pełnym*. Taki graf o n wierzchołkach oznaczany jest za pomocą symbolu K_n .

Definicja 15. Podgraf

Graf $G'(V', E')$ jest *podgrafem* grafu $G(V, E)$, jeśli zbiór wierzchołków V' jest niepustym podzbiorem zbioru V i zbiór krawędzi E' jest podzbiorem zbioru E . Jeśli $G' = (V', E')$ jest podgrafem G , to G jest jego *nadgrafem* (czasem spotykane jest określenie supergrafu [47]).

Definicja 16. Podgraf rozpinający

Podgraf $G' = (V', E')$, grafu $G = (V, E)$ dla którego zachodzi $V' = V$ i $E' \subseteq E$ jest nazywany jego *podgrafem rozpinającym*.

Definicja 17. Podgraf indukowany

Podgraf grafu G , indukowany niepustym zbiorem wierzchołków U , gdzie $U \subseteq V$, który zawiera wszystkie krawędzie ze zbioru E występujące pomiędzy wierzchołkami $u_i, u_j \in U$, jest nazywany *podgrafem indukowanym*.

Definicja 18. Dekompozycja grafowa

Podział grafu G na zbiór rozłącznych krawędziowo podgrafów $G'_1, G'_2, \dots, G'_r$, gdzie każda krawędź należy do dokładnie jednego z podgrafów G'_i , jest nazywany *dekompozycją grafową*.

2.1.2 Rodzaje grafów

Istnieje wiele różnych rodzajów grafów, które rozszerzają Definicję 1, poprzez modyfikacje istniejących reguł lub dodanie nowych. W tej sekcji przedstawione są typy grafów istotne dla metod porównywania zaprezentowanych w następnych rozdziałach.

Definicja 19. Graf dwudzielny

Grafem dwudzielnym nazywany jest graf, w którym istnieją dwa rozłączne zbiory wierzchołków $V = V_1 \cup V_2$, $V_1 \cap V_2 = \emptyset$ i zbiór krawędzi E występujących tylko pomiędzy wierzchołkami należącymi do różnych zbiorów $e_i = \{v_j \in V_1; v_k \in V_2\}$.

Za pomocą symbolu $K_{n,m}$ przedstawiany jest graf dwudzielny będący jednocześnie grafem pełnym, zawierającym wszystkie krawędzie występujące pomiędzy wierzchołkami ze zbiorów V_1 i V_2 , gdzie $n = |V_1|$ a $m = |V_2|$.

Definicja 20. Multigraf

Graf G jest nazywany *multigrafem*, jeśli zawiera on przynajmniej jedną parę wierzchołków v_i, v_j , pomiędzy którymi istnieją co najmniej dwie krawędzie. W przypadku digrafów mowa o istnieniu przynajmniej dwóch łuków (v_i, v_j) pomiędzy parą wierzchołków v_i, v_j [42].

Definicja 21. Drzewa

Drzewem nazywamy spójny, acykliczny graf, którego liczba krawędzi jest o jeden mniejsza od liczby jego wierzchołków.

$$|E| = |V| - 1 \quad (2.3)$$

W drzewie pomiędzy dowolnymi dwoma wierzchołkami występuje tylko jedna ścieżka. *Ukorzenione drzewo* posiada dokładnie jeden wierzchołek oznaczony jako korzeń, używany jako punkt odniesienia dla pozostałych wierzchołków [47]. W grafie nieskierowanym każdy z wierzchołków może pełnić rolę korzenia, natomiast w grafie skierowanym korzeniem może być tylko wierzchołek o zerowym stopniu wejściowym [42].

Definicja 22. Graf planarny

Grafami planarnymi nazywamy grafy, które można przedstawić w przestrzeni dwuwymiarowej w taki sposób, że żadne dwie krawędzie nie przecinają się. W swojej pracy [71] Kuratowski udowodnił, że graf planarny nie może zawierać podgrafu K_5 lub $K_{3,3}$.

Definicja 23. Graf etykietowany

Graf nazywamy *etykietowanym*, jeżeli posiada funkcję l przypisującą każdemu wierzchołkowi v ciąg znaków:

$$\forall_{v \in V} l(v) = 'tekst_etykiety_wierzchołka' \quad (2.4)$$

Definicja 24. Graf etykietowany krawędziowo

Graf nazywamy *etykietowanym krawędziowo*, jeżeli posiada funkcję l przypisującą każdej krawędzi e ciąg znaków:

$$\forall_{e \in E} l(e) = 'tekst_etykiety_krawędzi' \quad (2.5)$$

Z perspektywy tej rozprawy grafy planarne, dwudzielne i drzewa zasługują na szczególną uwagę. Jest to spowodowane tym, że dla danych rodzajów grafów opracowano wydajne algorytmy porównywania grafów. Wspomniane algorytmy zostały użyte jako baza lub inspiracja przy opracowywaniu nowych metod dla sieci Petriego przedstawionych w Rozdziałach 6-9.

2.2 Sieci Petriego

2.2.1 Historia powstania i rozwój sieci

Początki powstania sieci Petriego są fascynujące. Carl Adam Petri oparł ich reprezentację graficzną na szkolnym pomysle z 1939 roku, użytym do zapamiętania chemicznych reakcji. Wiele lat później powrócił do tego konceptu, który dostosował do postaci aplikowalnej dla teorii automatów.

Sam Carl Adam Petri był niemieckim matematykiem urodzonym 1926 roku w Lipsku. W latach 1949-1956 studiował matematykę na Politechnice w Hanowerze. W 1962 uzyskał stopień doktora na Politechnice w Darmstadt. Jego praca doktorska - *Kommunikation mit Automaten* (Komunikacja automatów) - uzyskała nagrodę dla najlepszego doktoratu w danym roku dla prac z tej dziedziny. Od lat 60-tych był związany z uniwersytetem w Bonn.

Jako naukowiec Carl Adam Petri interesował się różnymi obszarami nauki: od chemii, której nauczał w Anglii po wojnie, przez fizykę i matematykę, kończąc na informatyce. Swoje główne zainteresowanie skupił na teorii systemów, a dokładnie na systemach dyskretnych.

Profesor Silva określa go następującymi słowami:

(...) not a classical mathematician. Interested in the description of some real-life situations, he essentially worked at conceptual level, "opening windows" (i.e., providing foundations for new ways of thinking) to represent systems. This may be viewed as a profile less frequently exercised with success, than that of "theorem prover". In other words, he was much more conceptual than technical [115].

W czasie swojego życia Carl Petri za swoje naukowe osiągnięcia otrzymał szereg prestiżowych nagród jak np. Pierścień Wernera von Siemens, czy Order Holenderskiego Lwa. Carl Adam Petri zmarł w 2010 roku. Dorobek, który po sobie pozostawił, nie ogranicza się tylko do systemów dyskretnych. Wiele z zaproponowanych przez niego pomysłów zostało po latach docenionych. Bez względu na pozostałe sukcesy i dokonania na kartach historii zapisał się ze względu na swój pomysł z dzieciństwa, który rozrósł się do nieoczekiwanych przez nikogo rozmiarów.

Dla większości osób, które miały styczność z sieciami Petriego, są one nierozzerwalnie związane z ich reprezentacją graficzną w postaci dwudzielnego grafu skierowanego z tokenami przechowywanymi przez wierzchołki nazywane miejscami. Jest to swego rodzaju ironia, ponieważ autor intencjonalnie opóźnił publikację graficznej reprezentacji swojej metody:

I did not want the theory to appear as a graphical method instead of a mathematical attack on the then prevailing Automata Theory, based on arguments taken from modern Physics [39].

Kommunikation mit Automaten został początkowo opublikowany w języku niemieckim i czekał cztery lata na Holta, który na potrzeby armii amerykańskiej przetłumaczył pracę [94] i przyczynił się tym do jej popularyzacji w społeczności naukowej [21]. To opóźnienie prawdopodobnie było powodem niskiego zainteresowania w pierwszych latach po publikacji.

Sieci Petriego rozpoczęły swoje istnienie jako matematyczny formalizm o formie sieci warunkowej (condition/event net), z nieważonymi łukami i tylko jednym tokenem w całej sieci (1-safe net). W

późnych latach 60-tych wytworzyła się forma sieci z podziałem na miejsca i tranzycje, co wzmocniło ich możliwości reprezentacji procesów.

Lata 70-te przyniosły nowe zmiany. Zapis sieci Petriego został rozszerzony do postaci grafu ważonego, w którym wielokrotne łuki są przedstawione w formie nieujemnej liczby całkowitej – wagi. Po tej zmianie najnowszy wariant został nazwany *uogólnionymi sieciami Petriego* (ang. generalised Petri nets), podczas gdy poprzednie wersje określono mianem *oryginalnych sieci Petriego* (ang. original Petri nets). Wspomniane zmiany pozwoliły na symulowanie maszyny Turinga. Zaproponowano także pierwsze metody analizy oparte na osiągalności i pokryciu sieci. Dodatkowe rodzaje łuków, takie jak inhibitory zostały wprowadzone do teorii w 1976 roku [45].

Pod koniec lat 70-tych Jensen zdefiniował klasę sieci, dla których wprowadzone zostało rozróżnienie tokenów za pomocą funkcji kolorowania – kolorowane sieci Petriego [60]. W kolejnych latach zdefiniował kolejny wariant sieci Petriego wysokiego poziomu (ang. High-level Petri nets) stanowiący połączenie jego poprzedniego pomysłu z innymi ówczesznie istniejącymi wariantami sieci Petriego [61].

Nowa dekada przyniosła wzrost zainteresowania w kontekście systemów rozproszonych i reaktywnych, a sieci Petriego zaczęły przyciągać coraz więcej uwagi społeczności naukowej. Pierwsza konferencja *International Conference on Application of Petri Nets and Concurrency* (ICATPN) miała miejsce w 1980 roku. Od tamtego momentu obserwowany jest wyraźny wzrost zainteresowania i liczby publikacji na temat sieci Petriego.

W 1989 roku pojawiła się publikacja Tadao Muraty *Petri nets: Properties, analysis and applications* [83], która stanowi istotny moment w historii rozwoju sieci Petriego, grupując w jednej pracy dotychczasową wiedzę na temat własności i sposobu działania klasycznych sieci Petriego. W efekcie nadal cieszy się dużą popularnością i licznymi cytowaniami (ponad 7000 w roku 2024).

W latach dziewięćdziesiątych XX wieku pojawiły się liczne publikacje dotyczące modelowania i analizy sieci Petriego. Ta sytuacja spowodowała kolejny wzrost zainteresowania nimi oraz przyspieszyła proces nauki modelowania z użyciem sieci przez nowych użytkowników. W 1992 roku pojawiła się pierwsza wersja programu Integrated Net Analyzer (INA), będącego przez długi czas jednym z najlepszych ogólnodostępnych analizatorów sieci Petriego (program przestał być utrzymywany na początku trzeciej dekady XXI wieku). Z punktu widzenia tej rozprawy kolejną bardzo ważną datą był przełom lat 1993 i 1994, kiedy pojawiła się pierwsza propozycja zastosowania sieci Petriego w biologii [106]. Artykuł ten przedstawia reprezentacje sieci metabolicznych w postaci sieci Petriego. Jest to jeden z przykładów szerszego zjawiska, w którym sieci Petriego zaczęły wychodzić poza pierwotny obszar swoich zastosowań i znalazły zastosowanie oraz uznanie w nowych obszarach nauki.

Nowe tysiąclecie przyniosło dalszy rozwój dedykowanych aplikacji, jak *Charlie* [113], *Monalisa* [31], *TINA* [16] czy *Snoopy* [53]. Obserwuje się także wzrost liczby metod i narzędzi analitycznych. To właśnie w pierwszej dekadzie XXI wieku można było zaobserwować opracowanie szeregu metod dekompozycji (często stworzonych w ścisłym połączeniu z konkretnym obszarem zastosowań sieci Petriego, co czasami owocowało równoległym opracowaniem podobnych metod np. ADT i t-sieć).

Można również zaobserwować systematyczny wzrost liczby opracowywanych metod dedykowanych dla konkretnych rozszerzeń sieci Petriego, który przybierze na sile w drugiej dekadzie. Sieci klasyczne traktowane są często jako wprowadzenie do teorii sieci Petriego, po którym następuje przejście do nauki docelowego rozszerzenia np. czasowych sieci Petriego. Wciąż jednak istnieją obszary teorii sieci Petriego, które mimo upływu dekad nie zostały należycie zbadane, np. będąca tematem tej rozprawy

kwestia porównywania sieci Petriego, która doczekała się tylko jednej publikacji w 2012 roku [10]. Braki te są jednak systematycznie uzupełniane przez nowe pokolenia rozpoczynające swoją pracę naukową od sieci klasycznych [77, 108, 112].

Jako przykład trendów w obszarze badań nad sieciami Petriego panujących na początku trzeciej dekady XXI wieku, przywołana została publikacja z 2022 roku [40]. Na 10 zawartych w niej pozycji, dwie dotyczyły zastosowań klasycznych sieci Petriego, trzy czasowych, trzy kolorowanych i dwie stanowiące kombinacje stochastycznych modeli z czasowymi czy kolorowanych z czasowymi.

Główne rozszerzenia sieci Petriego

Jedną z największych zalet sieci Petriego jest to, że ich forma jest podatna na modyfikacje, takie jak dodanie nowych elementów, np. łuków blokujących czy ograniczeń, np. czasowe ograniczenia uruchomień. Przez siedemdziesiąt lat naukowcy rozszerzali definicję sieci Petriego, aby uzyskać coraz precyzyjniejsze narzędzie do reprezentacji procesów. Efektem jest wywodząca się z klasycznej definicji liczna rodzina rozszerzeń.

Jednym z bardziej popularnych rozszerzeń są **Kolorowane sieci Petriego**, zaproponowane przez Jensena w 1979 roku [60]. W porównaniu z klasyczną definicją, modyfikacja polegała na wprowadzeniu rozróżnienia pomiędzy tokenami znajdującymi się w sieci. Zostało to uzyskane za pomocą funkcji kolorującej, która przypisuje dla każdego tokenu wartość liczbową odpowiadającą jego *kolorowi* [116].

W przypadku **Ciągłych Sieci Petriego** radykalnej modyfikacji poddano jedną z podstawowych cech klasycznego wariantu. Token przestał być przedstawiany w postaci nieujemnej liczby całkowitej, a jest zapisywany za pomocą liczby rzeczywistej [29]. W rezultacie tej zmiany całkowitej przebudowie podlega analiza przestrzeni stanów, która staje się zbyt duża dla standardowych technik analizy znanych z klasycznego wariantu [29].

Pod terminem **Czasowych sieci Petriego** znajduje się kilka wariantów rozszerzeń, które w różny sposób podchodzą do kwestii ograniczeń czasowych jakim podlega uruchamiana tranzycja. Najbardziej rozpowszechnionymi wersjami są **Time Petri net (TPN)** [104] i **Timed Petri Net (DPN)** [81] określane także jako **Duration-time Petri net**. TPN wprowadza dla każdej aktywnej tranzycji dwa czasy graniczne. Pierwszy określa czas od którego może ona zostać uruchomiona, podczas gdy drugi określa czas po osiągnięciu którego musi zostać uruchomiona. Natomiast sieci DPN definiują czas po którym uruchomiona tranzycja wyprodukuje nowy token [101].

Stochastyczne sieci Petriego (SPN) to sieci w których czasy uruchomień tranzycji są definiowane przez zmienne losowe [70]. Alternatywna interpretacja SPN polega na porównaniu jej z czasowymi sieciami Petriego, gdzie SPN odpowiada DPN o losowym czasie produkcji tokenu przez uruchomioną tranzycję [14, 34]. Ze względu na wysoki stopień skomplikowania modeli SPN, podejścia analityczne dedykowane na nich skupiają się na symulacji i metodach heurystycznych [70].

Przypadki w których model sieci Petriego zawiera w sobie jednocześnie elementy z kilku samodzielnych rozszerzeń, np. czas produkcji tokenu (DPN) i rozróżnienie tokenów (sieci kolorowane), jest on określany jako przypadek **Hybrydowej sieci Petriego** [29].

2.2.2 Teoria

Definicja 25. Klasyczna sieć Petriego

Sieć Petriego jest opisana przez piątkę $PN = (T, P, F, W, M_0)$, gdzie:

- $T = \{t_1, t_2, \dots, t_m\}$ i $P = \{p_1, p_2, \dots, p_n\}$ są skończonymi, rozłącznymi zbiorami odpowiednio miejsc i tranzycji,
- $F \subseteq (P \times T) \cup (T \times P)$ jest skończonym zbiorem łuków,
- $W : F \rightarrow \mathbb{Z}^+$ jest funkcją wag dla każdego łuku,
- $M_0 : P \rightarrow \mathbb{N}$ jest funkcją przypisującą początkową liczbę tokenów do każdego miejsca [83].

Sieć Petriego ma graficzną formę w postaci dwudzielnego, ważonego digrafu. Składa się z dwóch rozłącznych zbiorów wierzchołków, które odpowiadają dwóm typom wierzchołków – miejscom i tranzycjom.

Typy wierzchołków

Miejsce jest typem wierzchołka graficznie reprezentowanego w postaci okręgu. Przechowuje ono nieujemną wartość całkowitą odpowiadającą liczbie tokenów. Wektor liczb całkowitych, w którym każda pozycja odpowiada miejscu, a wartość liczbie znajdujących się w nim tokenów, jest nazywany stanem M sieci (ang. net marking). Początkowe rozłożenie tokenów w sieci nazywamy stanem początkowym M_0 .

Drugim typem wierzchołka w sieci Petriego jest *tranzycja*, zazwyczaj graficznie reprezentowana przez prostokąt. Odpowiada ona za przepływ tokenów wewnątrz sieci. Odbywa się on za pomocą dwóch akcji, tj. *aktywacji* i *uruchomienia*.

W zależności od kontekstu naukowego w jakim powstaje sieć Petriego istnieją różne interpretacje ról jakie pełnią poszczególne typy wierzchołków. W Tabeli 2.1 przedstawiono kilka przykładów interpretacji.

Tabela 2.1: Różne interpretacje elementów sieci Petriego

Miejsca wejściowe	Tranzycje	Miejsca wyjściowe
Dane wejściowe	Obliczenia	Dane wyjściowe
Sygnał wejściowy	Procesor sygnałowy	Sygnał wyjściowy
Zasób konieczny do wykonania zadania	Warunek zwolnienia zasobu	Zwieńczenie w logice konkluzji
Bufor	Procesor	Bufor
Substrat/Enzym	Reakcja	Produkt
Pasywny komponent biologiczny	Aktywny komponent biologiczny	Pasywny komponent biologiczny

Dla każdego wierzchołka w sieci Petriego może zostać wyznaczony podzbiór wierzchołków połączonych z nim łukami wchodzącymi lub wychodzącymi:

$\bullet p_i$ – zbiór bezpośrednich tranzycji wejściowych dla miejsca p_i

$p_i^\bullet$ – zbiór bezpośrednich tranzycji wyjściowych dla miejsca p_i

$\bullet t_i$ – zbiór bezpośrednich miejsc wejściowych dla tranzycji t_i

$t_i^\bullet$ – zbiór bezpośrednich miejsc wyjściowych dla tranzycji t_i

$\bullet p_i^\bullet$ – zbiór bezpośrednich tranzycji wyjściowych i wyjściowych dla miejsca p_i

$\bullet t_i^\bullet$ – zbiór bezpośrednich miejsc wejściowych i wyjściowych dla tranzycji t_i

Wierzchołek v_i , dla którego $|\bullet v_i| > 0$ i $|v_i^\bullet| = 0$, jest zwany *wyjściowym* miejscem lub tranzycją. Wierzchołek v_i , dla którego $|v_i^\bullet| = 0$ i $|\bullet v_i| > 0$, nazywamy *wejściową* tranzycją lub miejscem.

Definicja 26. Macierz incydencji

Dla sieci Petriego macierz incydencji A jest macierzą o rozmiarze $n \times m$. Kolumny odpowiadają tranzycjom i ich zbiorom wychodzących i wchodzących łuków. Wiersze przedstawiają tę relację z perspektywy miejsc.

Dla sieci Petriego macierz incydencji A jest macierzą o rozmiarze $n \times m$ gdzie $a_{ij} = w(i, j)$ jeśli pomiędzy wierzchołkiem p_i a t_j występuje łuk (i, j) , lub $a_{ij} = -w(j, i)$ jeśli pomiędzy wierzchołkiem p_i a t_j występuje łuk (j, i) . W pozostałych przypadkach $a_{ij} = 0$.

Tranzycję t_i nazywamy *aktywną*, jeśli każde miejsce $p_{in} \in \bullet t_i$ posiada liczbę tokenów równą lub większą niż waga łuku łączącego p_{in} z t_i .

Aktywna tranzycja może, ale nie musi zostać *uruchomiona*. *Uruchomienie* jest atomowym procesem, podczas którego z każdego z miejsc $\bullet t_i$ pobierana jest liczba tokenów równa wadze łuków łączących je z uruchomioną tranzycją. W tym samym momencie generowane są nowe tokeny dla każdego miejsca z $t_i^\bullet$. Liczba wygenerowanych tokenów jest określona przez wagę łączącego łuku dane miejsce z uruchomioną tranzycją.

Typy łuków

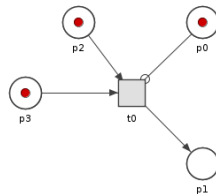
Wierzchołki w sieciach Petriego są połączone za pomocą łuków. Łuk łączy miejsce z tranzycją lub tranzycję z miejscem, nigdy dwa wierzchołki tego samego typu. Sieć Petriego pozwala na wielokrotne połączenia pomiędzy dwoma wierzchołkami. Przedstawiane są za pomocą jednego łuku z przypisaną nieujemną liczbą całkowitą, nazywaną wagą łuku. Istnieją także dwa typy łuków, które czasem są uznawane za część klasycznej definicji sieci Petriego, albo za samodzielne rozszerzenie sieci [93]:

- *Łuk odczytu* jest łukiem, który odpowiada dwukierunkowemu połączeniu pomiędzy miejscem p_r a tranzycją t_i . Uruchomiona tranzycja jednocześnie pobierze i zwróci token do miejsca p_r . Ponieważ uruchomienie jest operacją atomową, liczba tokenów w p_r nie ulegnie zmianie. Łuk odczytu jest zazwyczaj reprezentowany jako łuk o dwóch grotach (przykład zaprezentowano na Rysunku 2.2).



Rysunek 2.2: Sieć Petriego przedstawiającą działanie łuku odczytu łączącego tranzycję t_0 z miejscem p_0 przedstawiono na powyższym przykładzie. Lewa sieć przedstawia stan sieci przed uruchomieniem tranzycji t_0 . Prawa przedstawia stan sieci wyniku uruchomienia tranzycji t_0 , gdzie z miejsc p_2 i p_3 zostają pobrane tokeny, następnie utworzony zostaje nowy token w miejscu p_1 , a stan miejsca p_0 pozostaje bez zmian.

- *Inhibitor* lub *łuk blokujący*, jest łukiem, który łączy miejsce p_i z tranzycją t_j i pozwala na uruchomienie jej tylko w przypadku, gdy $M(p_i) = 0$. Inhibitor jest zazwyczaj reprezentowany w postaci łuku zakończonych kołem lub okręgiem wskazującym na tranzycję (przykład zaprezentowano na Rysunku 2.3).



Rysunek 2.3: Przykład przedstawia tranzycję t_0 połączoną z miejscem p_0 inhibitorem. Tak długo jak w miejscu p_0 znajduje się przynajmniej jeden token, tranzycja t_0 nie będzie aktywowana, bez względu na liczbę tokenów znajdującą się w miejscach p_2 i p_3 .

Oba typy łuków nie są reprezentowane w macierzy incydencji, ze względu na brak odpowiedniej wartości liczbowej, która przypisana do krotki a_{ij} , odpowiadałaby prawidłowemu uruchomieniu tranzycji t_j połączonej łukiem odczytu lub inhibitorem z miejscem p_i .

Niezmienniki

W sieciach Petriego istnieją dwa typy niezmienników bazujące na różnych typach wierzchołków.

t-niezmiennik jest wektorem x liczb naturalnych spełniającym równanie:

$$A \cdot x = 0 \quad (2.6)$$

Wsparcie t-niezmiennika x jest to zbiór $s(x) = \{t_j : x_j > 0, j = 1, 2, \dots, m\}$, gdzie m jest liczbą tranzycji. Należą do niego te tranzycje, którym odpowiadają dodatnie liczby w wektorze x .

t-niezmiennik wraz ze swoim wsparciem opisuje sekwencje uruchomień tranzycji, w wyniku którego osiągnany jest stan sprzed pierwszego uruchomienia. Za poprawny t-niezmiennik uznawany jest taki, w wyniku zajścia którego liczba tokenów w sieci nie uległa zmianie. W przypadku kiedy $Ax \neq 0$ wektor x rozróżnianych jest kilka rodzajów niezmienników:

- sub-niezmiennik – reprezentuje podproces, w którym suma tokenów w sieci zmniejsza się wraz z uruchomieniem wszystkich tranzycji ze wsparcia niezmiennika:

$$A \cdot x < 0 \quad (2.7)$$

- sur-niezmiennik – reprezentuje podproces, w którym suma tokenów w sieci zwiększa się wraz z uruchomieniem wszystkich tranzycji ze wsparcia niezmiennika:

$$A \cdot x > 0 \quad (2.8)$$

- non-niezmiennik – reprezentuje podproces niebędący poprawnym niezmiennikiem, gdzie wektor powstały w wyniku mnożenia A z x zawiera jednocześnie przynajmniej jedną liczbę dodatnią i przynajmniej jedną liczbę ujemną.

p-niezmiennik (w niektórych publikacjach s-niezmiennik) y jest wektorem liczb naturalnych odpowiadających liczbie odwiedzeń konkretnych miejsc przez token. Odpowiada on procesowi, w którym suma tokenów jest stała. p-niezmiennik spełnia warunek:

$$y \cdot A = 0 \quad (2.9)$$

t-niezmiennik jest nazywany *minimalnym*, jeśli w jego zbiorze tranzycji uzyskany za pomocą funkcji wsparcia, nie zawiera się żaden zbiór uzyskany dla innego niezmiennika.

2.2.3 Redukcja sieci

Nie istnieją powszechnie uznane reguły ustalające dokładnie jaką informację może przedstawiać pojedynczy wierzchołek. W efekcie taka sama informacja może być przedstawiona na różne sposoby, w zależności od interpretacji autora modelu. Problem ten jest szczególnie istotny z perspektywy niniejszej pracy, ponieważ może dojść do porównania dwóch modeli o różnych strukturach, gdzie każdy z nich w poprawny sposób odzwierciedla ten sam proces, ale robi to na różnych poziomach szczegółowości. Z tego powodu może okazać się koniecznym ujednolicenie sieci Petriego przed przeprowadzeniem właściwego porównania.

Kwestia upraszczania/redukcji modeli opartych o sieci Petriego została po raz pierwszy poruszona w latach siedemdziesiątych [72] i stanowiła próbę przeniesienia opracowanych wcześniej technik z teorii systemów [76] do sieci Petriego. Jako że nastąpiło to w końcowym okresie formalizowania postaci klasycznych sieci Petriego, z pewnym opóźnieniem pojawiły się prace dostosowujące redukcje do równoległych łuków czy łuków odczytu [73, 74].

Pod terminem redukcji sieci kryje się zbiór uproszczeń strukturalnych sieci Petriego, które zachowują podstawowe własności: niezmienniki i osiągalne stany [83]. Przykłady redukcji przedstawiono w Tabeli 2.2. Z perspektywy niniejszej rozprawy redukcje mogą zostać użyte do unifikacji porównywanych struktur (zrównania poziomów szczegółowości jakie reprezentują) oraz do zmniejszania czasu obliczeń poprzez operowanie na mniejszych modelach opartych na sieciach Petriego.

Tabela 2.2: Tabela przedstawia strukturalne redukcje sieci Petriego rozważane w pracach [74] i [83]. Każdą z redukcji przedstawiono za pomocą dwóch sieci, gdzie lewa odpowiada przypadkowi przed redukcją, a prawa po jej przeprowadzeniu.

1)		
2)		
3)		
4)		
5)		
6)		
7)		
8)		
9)		
10)		

Rozdział 3

Przegląd metod porównywania grafów i sieci

Niniejszy rozdział stanowi wprowadzenie do problemu porównywania grafów i sieci Petriego. Opisano w nim definicję izomorfizmu grafów stanowiącą fundament dla problemu porównywania grafów, oraz przedstawiono istniejące w literaturze metody porównywania, skupiając się na podejściach stanowiących podstawę dla metod porównywania sieci Petriego z Rozdziałów 6-9.

3.1 Izomorfizm

3.1.1 Teoria

Definicja 27. Izomorfizm grafów

Niech dane będą dwa grafy nieskierowane $G_1 = (V_1, E_1)$ i $G_2 = (V_2, E_2)$. Funkcja $f : V_1 \rightarrow V_2$ nazywana jest izomorfizmem grafów, jeżeli f jest funkcją wzajemnie jednoznaczną oraz dla każdej pary wierzchołków $x, y \in V_1$, $\{x, y\} \in E_1$ wtedy i tylko wtedy, gdy $\{f(x), f(y)\} \in E_2$.

Definicja 28. Automorfizm grafów

Automorfizmem grafu G_i nazywany jest jego izomorfizm z samym sobą. Oznacza to, że istnieje pewna permutacja wierzchołków ze zbioru V_i , która zachowuje relacje sąsiedztwa pomiędzy wierzchołkami [46].

Jeden z podstawowych sposobów podziału metod porównywania, dotyczy ich relacji do kwestii izomorfizmu. Rozróżniane jest dokładne dopasowanie grafów (ang. *exact graph matching*) i niedokładne dopasowanie grafów (ang. *inexact graph matching*). Główną różnicą jest to, że metody oparte o dokładne dopasowanie grafów operują na grafach posiadających tę samą liczbę wierzchołków, podczas gdy metody oparte o niedokładne dopasowanie grafów nie posiadają tego warunku. Tym samym niedokładne dopasowanie pozwala na porównanie grafów, w przypadku których wiadomo, że nie zachodzi izomorfizm, ale poszukiwana jest skala podobieństwa.

3.1.2 Historia

Każda poważna dyskusja o porównywaniu grafów powinna przynajmniej raz odnieść się do problemu izomorfizmu, mimo że przypadki, gdy dwa porównywane grafy są identyczne, nie należą do najczęstszych sytuacji w praktycznych zastosowaniach porównywania grafów. Nie zmienia to jednak faktu, że algorytmy do sprawdzania izomorfizmu stanowią bazę dla części metod porównywania oraz punkt odniesienia przy określaniu stopnia różnic pomiędzy dwoma grafami.

Pierwsza praca traktująca o problemie izomorfizmu grafów została opublikowana w 1957 roku dla problemu dopasowywania grafów molekularnych [105]. W latach 70-tych problem ten zaczął zdobywać zainteresowanie w środowisku naukowym, jednak mimo niesłabnącego zainteresowania przynależność problemu izomorfizmu grafów do klasy NP-zupełnych pozostaje otwarta [48, 134]. Efektem tego jest brak wielomianowych dokładnych algorytmów wyznaczania izomorfizmu dla grafów prostych. Niespodziewanie, w 2015 roku pojawiły się doniesienia o opracowaniu pseudo-wielomianowego algorytmu dla nieskierowanych grafów [7], co na przestrzeni dziesiątek lat nieskutecznych prób stanowiłoby znaczący przełom. W pracy stosunkowo szybko odnaleziono i skorygowano pewne błędy po czym nowa wersja została zaproponowana w 2017 [54]. Niestety do drugiego kwartału 2024 praca nie doczekała się publikacji i wciąż powoduje dyskusję pośród naukowców zainteresowanych tematem izomorfizmu grafów [43, 130, 132].

Tabela 3.1: Porównanie złożoności czasowej algorytmów sprawdzających izomorfizm dla wybranych klas grafów

Typ grafu	Złożoność czasowa	$O(n)$	Autorzy
planarny	wielomianowa	$O(V \log V)$	Hopcroft i Tarjan [57]
drzewo	wielomianowa	$O(v^2)$	Campbell [23]
skierowany	wielomianowa	$O(v^4)$	Yang i May [133]
o ograniczonym stopniu	wielomianowa	$O(\sqrt{v \log v})$	Luks [78]

Podczas gdy poszukiwania efektywnego algorytmu rozwiązującego problem izomorfizmu dla grafów prostych nadal trwają, obserwowany jest stały trend opracowywania dokładnych metod dedykowanych dla konkretnych klas grafów np. grafów planarnych [57]; drzew [23]; grafów o ograniczonym stopniu [78] (Tabela 3.1). Dla przypadków gdzie jest akceptowalna przybliżona skala podobieństwa grafów, istnieją liczne algorytmy heurystyczne [19, 26, 27, 75].

3.2 Metody porównywania grafów

Porównywanie grafów występuje w literaturze pod wieloma angielskimi nazwami np. *network alignment*, *graph matching*, *graph comparison*. Wszystkie one reprezentują metody używane do określania stopnia podobieństwa lub różnic występujących pomiędzy dwoma grafami. W 1963 Sussenguth opracował pierwszy algorytm do porównywania małych grafów chemicznych, bazujący na odnajdywaniu izomorfizmu [119]. W artykule [32] zaproponowano podział metod porównywania, który przedstawiono w Tabeli 3.2.

Tabela 3.2: Podział metod porównywania na poszczególne rodzaje według Emmert-Streiba [32]

SIECI DETERMINISTYCZNE	SIECI LOSOWE
Miary globalne	Miary globalne
<i>Dokładne dopasowanie grafów</i>	Grafowa odległość edycyjna
Miary oparte o izomorfizm	Dopasowanie sieci
<i>Niedokładne dopasowanie grafów</i>	Dopasowanie sieci z użyciem zewnętrznych informacji
Grafowa odległość edycyjna	Miary lokalne
Gramatyki grafowe	Dystans znakowy
Metody iteracyjne	Dystrybucja stopni grafletów
Jądra grafowe	Metody wolnego dopasowania
Miary lokalne	Metody oparte o testy Hipotez
Miary oparte o izomorfizm	Dopasowanie sieci z użyciem zewnętrznych informacji
Miary oparte o podsieci	
Miary informacyjno-interpretacyjne	
Drzewa	
Dystans drzewny	
Dystans znakowy	

3.2.1 Dystans grafowy

Pierwszą opracowaną metryką służącą do porównania grafów jest dystans grafowy, zaproponowany przez Bohdana Zelinkę w 1974 [141]. Bazuje on na pracy Vizinga [127] i jego podejściu do rozwiązywania problemu izomorfizmu.

Definicja 29. Dystans grafowy

Dystansem grafowym pomiędzy grafem G_1 , a G_2 nazywamy liczbę naturalną d , która odpowiada liczbie wierzchołków po których usunięciu lub dodaniu graf G_1 będzie izomorficzny z G_2 [141].

Definicja 30. Krawędziowy dystans grafowy

Krawędziowy dystans grafowy pomiędzy grafem G_1 , a G_2 nazywamy liczbę naturalną d^k , która odpowiada liczbie krawędzi po których usunięciu lub dodaniu graf G_1 będzie izomorficzny z G_2 [142].

Dystans grafowy $d(G_1, G_2)$ oraz krawędziowy $d^k(G_1, G_2)$ są metrykami i spełniają następujące własności [8]:

(1) Nieujemność

$$d(G_1, G_2) \geq 0 \quad (3.1)$$

(2) Tożsamość nierozróżnialnych elementów

$$d(G_1, G_2) = 0 \Leftrightarrow G_1 = G_2 \quad (3.2)$$

(3) Symetria

$$d(G_1, G_2) = d(G_2, G_1) \quad (3.3)$$

(4) Nierówność trójkąta

$$d(G_1, G_2) + d(G_2, G_3) \geq d(G_1, G_3) \quad (3.4)$$

W oryginalnej formie dystans grafowy został zaproponowany jak metoda idealnego dopasowania. Dopiero w 1990 roku w artykule [63] zaproponowana została modyfikacja definicji dystansu pozwalająca na porównywanie dwóch grafów o różnej liczbie wierzchołków. Ponadto zaprezentowane zostały dwie metryki określające dystans. W zależności od typów porównywanych grafów należy zastosować odpowiednią z dwóch wspomnianych metryk:

$$d_M^A(G_1, G_2) = \max\{n(G_1), n(G_2)\} - \max\{n(H)\} \quad (3.5)$$

gdzie H jest największym wspólnym podgrafem dla G_1 i G_2 , a $n()$ funkcją zwracającą liczbę wierzchołków w danym grafie.

$$d_M^B(G_1, G_2) = \min\{n(M)\} - \min\{n(G_1), n(G_2)\} \quad (3.6)$$

gdzie M jest najmniejszym nadgrafem zawierającym w sobie G_1 i G_2 jako podgrafy.

3.2.2 Największy wspólny podgraf

Kolejnym podejściem do porównywania grafów jest znalezienie *największego wspólnego podgrafu* (NWP). Największym wspólnym podgrafem grafów G_1 oraz G_2 , nazywany jest maksymalnej wielkości podgraf H występujący jednocześnie w grafie G_1 oraz G_2 . Jest to znany problem NP-zupełny [80, 90], podczas gdy znalezienie *największego wspólnego indukowanego podgrafu* (NWIP) reprezentuje inny problem, który jest z kolei problemem NP-trudnym [38]. Niestety, dla dużych grafów te podejścia nie są wystarczająco wydajne. Dlatego też podobnie jak w przypadku problemu izomorfizmu grafów można zaobserwować duże zainteresowanie algorytmami przybliżonymi.

Powiązanie pomiędzy *NWP* a dystansem grafowym jest łatwe do zaobserwowania w równaniu 3.5, gdzie *NWP* stanowi punkt odniesienia, od którego zliczane są wierzchołki reprezentujące dystans.

3.2.3 Grafowa odległość edycyjna

Grafowa odległość edycyjna (ang. *Graph edit distance* GED) jest miarą podobieństwa, będącą adaptacją dobrze znanej znakowej odległości edycyjnej (ang. *String Edit Distance*) do teorii grafów. Początkowo zaproponowana przez Sanfeliu i Fu w 1983 [109]. Porównując jej działanie z dystansem grafowym, grafowa odległość edycyjna nie reprezentuje liczby elementów, które należy zmienić aby graf G_1 był izomorficzny z grafem G_2 , ale sumaryczny koszt minimalnego zbioru operacji o_i transformujących graf G_1 w graf G_2 . Obliczenie GED jest problemem NP-trudnym [144].

Podstawowymi operacjami o_i jakie można przeprowadzić na grafie są:

- $(v_1 \rightarrow v_2)$ - podmiana wierzchołka,
- $(e_1 \rightarrow e_2)$ - podmiana krawędzi,
- $(v_1 \rightarrow \emptyset)$ - usunięcie wierzchołka,
- $(e_1 \rightarrow \emptyset)$ - usunięcie krawędzi,
- $(\emptyset \rightarrow v_2)$ - dodanie wierzchołka,
- $(\emptyset \rightarrow e_2)$ - dodanie krawędzi [22].

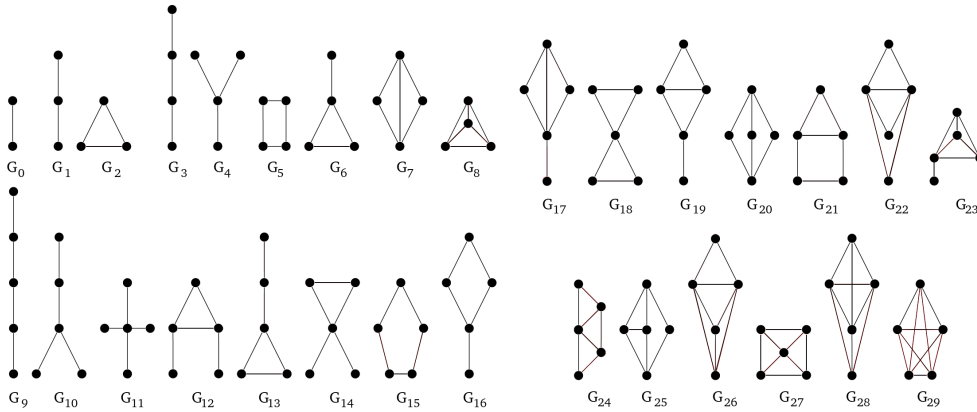
Niech $G_1 = (V_1, E_1)$ będzie początkową postacią grafu, a $G_2 = (V_2, E_2)$ końcową. Grafowa odległość edycyjna $d^{GED}(G_1, G_2)$ pomiędzy grafami G_1 and G_2 jest zdefiniowana za pomocą następującego równania:

$$d^{GED}(G_1, G_2) = \min_{\{o_1, \dots, o_k\} \in T(G_1, G_2)} \left\{ \sum_{o_i \in \{o_1, \dots, o_k\}} c(o_i) \right\} \quad (3.7)$$

gdzie $T(G_1, G_2)$ jest zbiorem wszystkich możliwych transformacji grafu G_1 w graf G_2 , pojedyncza transformacja grafu jest przedstawiana za pomocą zbioru operacji $\{o_1, \dots, o_k\}$, a $c(o_i)$ reprezentuje funkcję kosztu operacji o_i [109, 114].

3.3 Graflety

Za samodzielną grupę metod porównywania sieci bazujących na znajdowaniu lokalnych podobieństw [32], uznawane są metody oparte o badanie rozmieszczenia małych nieizomorficznych podsieci, nazywanych *grafletami*. Istotną różnicą w porównaniu do innych wspomnianych wcześniej podejść porównywania jest to, że od początku graflety są przeznaczone do zastosowania w przypadkach porównywania dużych sieci o rozmiarach od kilku tysięcy do kilkudziesięciu tysięcy wierzchołków.



Rysunek 3.1: Zbiór wszystkich nieskierowanych grafletów o wielkości od 2 do 5 wierzchołków.

3.3.1 Historia

Pierwszy raz definicja grafletów wraz z propozycją zastosowania pojawia się w artykule [99] Nataszy Prżulij. Reprezentują one małe podsieci, w których rozmieszczenie w porównywanych sieciach podlega analizie. Dla grafów nieskierowanych istnieje 30 nieizomorficznych podgrafów o rozmiarach od 2 do 5 wierzchołków, które przedstawiono na Rysunku 3.1. Inspiracją dla grafletów były *motywy* oraz bazujące na nich metody porównywania [82]. Z dzisiejszej perspektywy można stwierdzić, że zdefiniowanie orbit znacząco wpłynęło na rozwój grafletów. Orbitą nazywamy zbiór wierzchołków występujących w graflecie, które można przekształcić w siebie nawzajem w wyniku automorfizmu jaki występuje w danym graflecie. Jednocześnie wraz z ich wprowadzeniem w 2007 roku, opracowano nową bazującą na nich metodę porównywania, nazywaną Rozkładem Stopni Grafletów (ang. *Graphlet Degree Distribution* GDD) i jej znormalizowaną postać Uzgodnionego Rozkładu Stopni Grafletów (ang. *Graphlet Degree*

Distribution Agreement GDDA). Od tego momentu obserwuje się znaczący wzrost zainteresowania grafletami i metodami bazującymi na nich. Powodu takiej sytuacji należy szukać w zdefiniowaniu orbit oraz potencjału, jaki wniosły dla tej rodziny metod porównywania.

Kwestia znalezienia wszystkich grafletów w porównywanych sieciach stanowi pierwszy i jednocześnie najbardziej złożony obliczeniowo etap w porównywaniu sieci za pomocą grafletów [2, 97]. Już przypadku sieci interakcji międzyproteinowych, które były pierwotnym typem sieci dla którego zastosowano graflety, średnia liczba grafletów występująca w sieciach tego typu wymusiła opracowanie heurystycznych podejść zliczania grafletów [18, 41]. Jednocześnie prowadzone były prace nad algorytmami pozwalającymi na wyznaczanie grafletów większych rozmiarów niż 5 wierzchołków, zaproponowanych przez Pržulij. Przykładem takiego algorytmu jest np. G-trie będący zaadaptowanym algorytmem wyznaczania motywów [5]. Wraz z zastosowaniem grafletów większych rozmiarów, pojawia się problem wysokiej ich liczby. Skala przyrostu jest tak duża, że dla grafletów o rozmiarze 8 i większych proponowane są heurystyczne metody ich wyznaczania [49].

Z perspektywy niniejszej rozprawy znaczną obawę rodził problemem stabilności wyników zwracanych przez metodę GDDA zaobserwowany w grafach o niskiej gęstości [50, 107]. Gęstością grafu określa się stosunek liczby łuków względem liczby wierzchołków. Powodem obaw był fakt, iż większość modeli biologicznych utworzonych za pomocą sieci Petriego reprezentuje sieci o stosunkowo niskiej gęstości. Wymusiło to przeprowadzenie testów badających zakres i poziom niestabilności GDDA dla modeli biologicznych przedstawionych za pomocą sieci Petriego, co zostało opisane w artykule [122] oraz Rozdziale 6.

Graflety i ich możliwości zostały zauważone i zaadaptowane do nowych obszarów, takich jak: sieć WWW [110], sieci ekologiczne [28], sieć handlu światowego (skierowany wariant) [110] czy sieci metaboliczne (pn-graflety) [120].

3.3.2 Digraflety

Graflety zostały pierwotnie zdefiniowane dla grafów nieskierowanych, którym odpowiadały sieci interakcji międzyproteinowych. Zastosowanie nieskierowanych grafletów na modelach sieci reprezentowanych przez grafy skierowane w większości przypadków skutkuje niewystarczającymi wynikami. Powodem jest utrata kluczowej informacji na temat kierunku, co może podważyć sens przeprowadzanego porównania. W związku z czym konieczne było odpowiednie zmodyfikowanie grafletów po przez zamianę krawędzi na łuki i ponowne wyznaczenie wszystkich nieizomorficznych podsieci.

Pomysł adaptacji grafletów do grafów skierowanych pojawił się jednocześnie w dwóch niepowiązanych ośrodkach, co można zaobserwować na podstawie dwóch artykułów, tj. zespołu profesor N. Pržulj [110] i zespołu profesora F. Silvy [5]. Oba definiują taki sam zbiór skierowanych grafletów (digrafletów), przy czym ostatni rozważa dodatkowo przypadki z dwukierunkowymi łukami.

Istnieje 40 nieizomorficznych digrafletów o rozmiarze do 4 wierzchołków, które nie posiadają łuków dwukierunkowych [110]. W przypadku digrafletów posiadających łuki dwukierunkowe, mowa o 214 grafletach o rozmiarze do 4 wierzchołków [5].

3.3.3 Metryki grafletowe

Aby przy porównywaniu grafów w pełni skorzystać z możliwości jakie daje użycie grafletów, konieczne było opracowanie odpowiednich miar podobieństwa bazujących na nich. Część z nich została zaadop-

towana z innych pokrewnych typów grafów, np. motywy z sieci interakcji międzyproteinowych [82], podczas gdy pozostałe powstawały specjalnie dla grafletów [3, 98].

Dystans Względnej Częstotliwości Graflightów (ang. Relative Graphlet Frequency distance RGF) jest pierwszą opracowaną miarą używającą graflightów do porównywania sieci. Bazuje ona na metodzie stworzonej dla motywów w grafów losowych [82]. W celu wyznaczenia RGF konieczne są następujące obliczenia:

$$T(G) = \sum_{i=0}^{29} N_i(G) \quad (3.8)$$

gdzie $N_i(G)$ zwraca liczbę wystąpień i -tego graflightu (ze zbioru przedstawionego na Rysunku 3.1) w grafie G .

$$F_i(G) = -\log(N_i(G)/T(G)) \quad (3.9)$$

gdzie $F_i(G)$ jest normalizacją wartości $N_i(G)$.

$$D(G, H) = \sum_{i=0}^{29} |F_i(G) - F_i(H)| \quad (3.10)$$

gdzie $D(G, H)$ określa sumę znormalizowanych dystansów pomiędzy liczbą wystąpień każdego graflightu w dwóch porównywanych sieciach G i H .

W przypadku sieci interakcji międzyproteinowych, dla których pierwotnie został opracowany RGF, zwracane przez niego wyniki okazały się niewystarczające, co stanowiło impuls do opracowania nowych metod.

Rozkład Stopni Graflightów (ang. *Graphlet Degree Distribution* GDD) reprezentuje podejście zaproponowane w [98]. Istotną zmianą jest wprowadzenie pojęcia orbity wraz przeniesieniem uwagi z graflightu na znajdującą się w nim orbitę.

Orbitą nazywamy zbiór wierzchołków w ramach graflightu, dla których zachodzi relacja automorfizmu. Automorfizm reprezentuje specjalny przypadek izomorfizmu, gdy graf jest izomorficzny ze samym sobą. Zbiór wszystkich automorfizmów grafu G jest oznaczany $Aut(G)$. Dla $v \in V$ orbitą tego wierzchołka jest zbiór $Orb(v) = \{u \in V : u = f(v), f \in Aut(G)\}$.

Pierwszym krokiem przy wyznaczaniu macierzy GDD jest wyznaczenie macierzy wystąpień Fr_G :

$$Fr_G = \begin{bmatrix} f_{0,0} & f_{0,1} & \dots & f_{0,m} \\ f_{1,0} & f_{1,1} & \dots & f_{1,m} \\ \vdots & \vdots & \ddots & \vdots \\ f_{n,0} & f_{n,1} & \dots & f_{n,m} \end{bmatrix} \quad (3.11)$$

gdzie n jest liczbą wierzchołków w sieci, m jest liczbą orbit, a $f_{i,j}$ jest liczbą wystąpień orbity j w wierzchołku i .

$$GDD_G = \begin{bmatrix} d_G^0(1) & d_G^0(2) & \dots & d_G^0(\alpha) \\ d_G^1(1) & d_G^1(2) & \dots & d_G^1(\alpha) \\ \vdots & \vdots & \ddots & \vdots \\ d_G^m(1) & d_G^m(2) & \dots & d_G^m(\alpha) \end{bmatrix}. \quad (3.12)$$

gdzie $d_G^j(k)$ oznacza liczbę k -wystąpień j -jej orbity we wszystkich wierzchołkach grafu G , a α oznacza największą liczbę k -wystąpień spośród wszystkich orbit.

Uzgodniony Rozkład Stopni Grafięć (ang. *Graphlet Degree Distribution Agreement* GDDA) został zaproponowany równocześnie z GDD [98] i stanowi jego uzupełnienie. Aby obliczyć GDDA należy dla każdej z pozycji w macierzy GDD wyznaczyć wartość $S_G^j(k)$:

$$S_G^j(k) = \frac{d_G^j(k)}{k} \quad (3.13)$$

która odpowiada skalowanej wartości $d_G^j(k)$. Następnie wyznaczana jest wartość $N_G^j(k)$ na podstawie wzoru:

$$N_G^j(k) = \frac{S_G^j(k)}{\sum_{k=1}^{\alpha} S_G^j(k)}. \quad (3.14)$$

gdzie obliczana jest normalizacja dystrybucji j -tej orbity.

Następnie dla grafów G i H dystans j -ty orbity bazujący na ich znormalizowanych dystrybucjach jest definiowany:

$$D^j(G, H) = \frac{1}{\sqrt{2}} \left(\sum_{k=1}^{\alpha} [N_G^j(k) - N_H^j(k)]^2 \right)^{\frac{1}{2}}. \quad (3.15)$$

Jest to poprawiona wersja wzoru przedstawiona w erracie artykułu [96].

Wynik cząstkowy dla j -tej orbity przyjmuje wartość z przedziału $[0, 1]$, gdzie 0 oznacza identyczny rozkład dystrybucji, 1 rozłączny (niepokrywające się) rozkład, a wartości pośrednie częściową zgodność.

$$A^j(G, H) = 1 - D^j(G, H). \quad (3.16)$$

Końcowy wynik odpowiadający wartości GDDA jest średnią sumą cząstkowych wyników $A^j(G, H)$:

$$A(G, H) = \frac{1}{73} \sum_{j=0}^{72} A^j(G, H). \quad (3.17)$$

Zaobserwowano w [107] i potwierdzono w [50], że GDDA zachowuje się „niestabilnie” w grafach o niskiej gęstości - niskim stosunku liczby krawędzi do liczby wierzchołków. Niestabilność objawia się okresowym spadkiem średniej wartości podobieństwa.

Oprócz przedstawionych powyżej RGF i GDDA, w literaturze występują także inne metryki porównywania sieci z użyciem grafięć. NetDis została zaproponowana w [3] jako odpowiedź dla GDDA i jej problemów w grafach o niskiej gęstości. Bazuje ona na znajdowaniu rozkładu grafięć, ale w

przeciwieństwie do RGF nie w ramach całej sieci, a w ramach wszystkich podsieci o zadanej wielkości jakie występują w sieci. Niestety przeprowadzone testy wykazały niską skuteczność metody po wstępnej adaptacji do sieci Petriego, w wyniku czego zaprzestano dalszych prac nad nią.

Efektywność metody NetDis w zastosowaniu do sieci interakcji międzyproteinowych, została poddana pod wątpliwość w pracy [136]. Stwierdzono w niej mniejszą efektywność od przedstawianej przez autorów NetDis w artykule [3]. Metoda została uznana za słabszą od Korelacyjnego Dystansu Grafiowego (ang. *Graphlet Correlation Distance* GCD) [135] i porównywalną pod względem możliwości z RGF i GDDA. W ramach niniejszej rozprawy nie zostały podjęte prace nad dostosowaniem GCD do operowania na sieciach Petriego, ze względu na potrzebę opracowania nowych tablic zależności międzygrafiowych dla grafiów dedykowanych do sieci Petriego.

3.4 Porównywanie przez niezmienniki

Niezmienniki razem ze swoimi wsparciami odpowiadają powtarzalnym i stabilnym podprocesom jakie zachodzą w modelowanym systemie przedstawionym przez sieci Petriego. Wyjątkiem są sub- i sur-niezmienniki które reprezentują procesy w wyniku których liczba tokenów z perspektywy globalnego stanu sieci, kolejno większa się lub zmniejsza. Bez względu na typ, niezmienniki mogą reprezentować różne rodzaje struktur, od prostych ścieżek, cykli i drzew po bardziej skomplikowane struktury.

3.4.1 Historia rozwoju

Pierwsza dedykowana metoda do porównywania sieci Petriego została zaproponowana w 2012 roku przez zespół naukowców z uniwersytetów Padova i Valenzia [10]. Co ciekawe, zastosowanie do sieci Petriego nie było głównym celem, jaki przyświecał autorom. Poszukiwali oni nowych metod porównywania sieci metabolicznych. Istnieją wyraźne podobieństwa pomiędzy sieciami metabolicznymi a modelami przedstawionymi za pomocą sieci Petriego: macierz stechiometryczna dla sieci metabolicznych odpowiada macierzy incydencji w sieciach Petriego, a t-niezmienniki odpowiadają przepływowi metabolitów (ang. fluxes). Możliwość transformacji pomiędzy tymi typami sieci jest znana od 1994 roku [106]. Do czasu badań przeprowadzanych w ramach niniejszej dysertacji, metoda porównująca sieci za pomocą niezmienników była jedyną opublikowaną metodą dedykowaną dla sieci Petriego [10].

Wspominana metoda reprezentuje kombinację dwóch istniejących już podejść do porównywania sieci metabolicznych. Pierwsze podejście do porównania sieci metabolicznych bazuje na porównaniu wartości Komisyjnego Numeru Enzymu (ang. Enzyme Commission Number - EC) [128] przypisanych do enzymów reprezentowanych przez wierzchołki. Drugie podejście bazuje na porównywaniu przepływów metabolitów, reprezentowanych przez t-niezmienniki. Wynik końcowy jest sumą dwóch składowych wartości zwracanych przez obie podmetody, których wzajemny stosunek jest sterowany parametrem ustalonym przez użytkownika.

Pomysł ten nie zrewolucjonizował porównywania sieci metabolicznych, ale został odnotowany jako kolejne dostępne podejście. Co istotne, autorzy opracowali jednocześnie aplikację *CoMeta*, która pozwala na porównywanie modeli sieci metabolicznych, za pomocą transformacji ich do sieci Petriego w celu określenia ich podobieństwa. Tym samym stając się pierwszym programem pozwalającym na porównywanie sieci Petriego [11].

3.4.2 Obszary zastosowań

Zastosowanie tej metody porównywania w obszarze innym niż sieci metaboliczne natrafia na szereg ograniczeń spowodowanych wymaganiami dla obu składających się na nią podmetod. Numeracja EC pozwala na porównanie dwóch enzymów/wierzchołków, jednak obecność odpowiedników numeracji EC w innych obszarach gdzie występują modele bazujące na sieciach Petriego jest mało prawdopodobna. Próba zastosowania zamiennika w postaci porównania etykiet wierzchołków np. metodą Levenshteina, będzie zwracać znacząco gorsze wyniki. W porównaniu z pierwszą podmetodą, druga bazująca na t-niezmiennikach nie wykazuje tak dużego przywiązania do sieci metabolicznych i może być skutecznie aplikowana do innych modeli. Jej ograniczeniem jest natomiast możliwość zajścia wykładniczego wzrostu liczby cząstkowych wektorów podczas wyznaczania zbioru minimalnych niezmienników, co blokuje możliwość jej wykorzystania.

Podmetoda bazująca na niezmiennikach może zostać samodzielnie użyta do detekcji zmian w zbiorze minimalnych niezmienników po wykonaniu operacji blokowania na wybranych wierzchołkach. W takim przypadku dochodzi do porównania oryginalnej sieci z blokową jako dwóch osobnych modeli. Przykład takiego zastosowania został zaprezentowany w [35].

3.4.3 Metoda składowa bazująca na wartościach EC

Pierwsza metoda składowa wyznacza częściowy wynik (R_score) zliczając reakcje homologiczne występujące w sieci. Do tego wykorzystuje numer EC, które zostały opracowane w International Union of Biochemistry and Molecular Biology (IUBMB), reprezentowany za pomocą czterech liczb:

- d_1 określa główną klasę enzymu, na podstawie ogólnego typu reakcji chemicznej, którą katalizuje.
- d_2 określa podklasę, uwzględniając bardziej szczegółowy typ reakcji.
- d_3 określa podpodklasę, zazwyczaj opisując specyficzny substrat lub grupę substratów, na które działa enzym.
- d_4 to numer seryjny, który jest unikalny dla konkretnego enzymu w danej podpodklasie.

Enzym e jest przedstawiany za pomocą następującej uporządkowanej czwórki:

$$e = (d_1, d_2, d_3, d_4) \quad (3.18)$$

Podobieństwo enzymów opisanych za pomocą numeru EC jest określane wzorem:

$$S(e, e') = \frac{\max\{(d_1, d_2, \dots, d_i), (d'_1, d'_2, \dots, d'_i)\}}{4} \quad (3.19)$$

Jako przykład posłużą dwa enzymy: Arginaza ($e = 3.5.3.1$) i Creatinaza ($e' = 3.5.3.3$), dla których podobieństwo $S(e, e')$ wynosi 0,75 [10].

3.4.4 Metoda składowa bazująca na niezmiennikach

Druga metoda składowa I_score bazuje na porównywaniu t-niezmienników. Jako że t-niezmiennik jest tylko wektorem liczb, do pełnego porównania niezmienników należących do różnych sieci konieczne

jest wyznaczanie i użycie wsparcia dla każdego niezmiennika. Autorzy tej metody rozważali użycie jedynie poprawnych minimalnych t-niezmienników.

3.4.5 Algorytm

Porównanie następuje poprzez znalezienie dwóch dystansów przedstawionych przez I_score odpowiadający porównaniu niezmienników i R_score odpowiadający różnicom pomiędzy Komisijnymi Numerami Enzymów znajdujących się w porównywanych sieciach. Do wyznaczenia wartości I_score oraz R_score stosowany jest indeks Sørensen S_index [118]:

$$S_index(X_1, X_2) = \frac{2|X_1 \cap X_2|}{|X_1| + |X_2|} \quad (3.20)$$

Alternatywą jest użycie indeksu Tanimoto T_index :

$$T_index(X_1, X_2) = \frac{|X_1 \cap X_2|}{|X_1| + |X_2| - |X_1 \cap X_2|} \quad (3.21)$$

Różnicą pomiędzy I_score , a R_score jest zawartość zbiorów X_1 i X_2 na których operuje wybrany indeks. W przypadku I_score odpowiadają zbiorom minimalnych t-niezmienników, a w przypadku R_score multizbiory numerów EC. Dwa enzymy są uznawane za podobne i trafiają do wspólnego zbioru jeśli wartość $S(e, e')$ ze wzoru 3.19 osiągnie ustalony wcześniej minimalny poziom. Do ustalenia czy dwa t-niezmienniki są podobne autorzy [10] stosują osobną heurystykę.

Obie wartości I_score i R_score są transformowane do postaci dystansów:

$$d_I(PN_1, PN_2) = 1 - I_score(X_1, X_2) \quad (3.22)$$

$$d_R(PN_1, PN_2) = 1 - R_score(E_1, E_2) \quad (3.23)$$

Ostatnim krokiem jest połączenie obu wartości w jeden dystans. Wpływ poszczególnych składowych jest sterowany przez użytkownika za pomocą parametru α , gdzie wartość 0 będzie oznaczać wykorzystanie tylko dystansu $d_I(PN_1, PN_2)$, wartość 1 będzie oznaczać wykorzystanie tylko dystansu $d_R(PN_1, PN_2)$.

$$d_D(PN_1, PN_2) = \alpha d_R(PN_1, PN_2) + (1 - \alpha) d_I(PN_1, PN_2) \quad (3.24)$$

gdzie $d_D(PN_1, PN_2)$ określa składowy dystans pomiędzy dwoma sieciami metabolicznymi przedstawionymi za pomocą sieci Petriego. Możliwe jest także wyznaczenie dystansu pomiędzy dwoma organizmami opisywanymi przez sieci metaboliczne znajdujące się w tych organizmach.

$$d_D(O_1, O_2) = \frac{\sum_{j=1}^n d_d^j(PN_j^1, PN_j^2)}{n} \quad (3.25)$$

gdzie $d_D(O_1, O_2)$ odpowiada wyznaczonemu dystansowi pomiędzy dwoma organizmami O_1 i O_2 , zawierającymi n sieci metabolicznych reprezentowanych przez sieci Petriego.

Z perspektywy niniejszej rozprawy skupiającej się na porównywaniu dwóch sieci Petriego bez wykorzystywania zewnętrznych baz danych jak np. bazy enzymów EC, to wynik w postaci dystansu $d_I(PN_1, PN_2)$ z równania 3.22 przedstawia oczekiwany stopień podobieństwa.

Rozdział 4

Dekompozycja sieci Petriego

4.1 Wstęp

Podział sieci na mniejsze struktury posiadające pożądane charakterystyki nie jest nowym podejściem analitycznym w teorii sieci Petriego. Od początku lat dziewięćdziesiątych obserwuje się opracowywanie technik bazujących na istniejących wcześniej dekompozycjach grafowych. W zależności od środowiska, w którym używane są sieci Petriego (biologia, automatyka, informatyka), różne elementy takie jak miejsca, tranzycje, czy wielokrotne łuki, będą na potrzeby analizy uznawane za kluczowe, co wpływa na kształt uzyskiwanych podsieci.

Zaprezentowane w tym rozdziale dekompozycje współdzielą szereg reguł, co wpłynęło na wybranie ich spośród dostępnych metod jako nośnika informacji w metodzie porównywania przedstawionej dokładnie w Rozdziale 9.

Łatwo pośród nich wydzielić metody bazujące na niezmiennikach, które z kolei można podzielić na stworzone dla t- lub p-niezmienników.

4.2 Dekompozycja sieci Petriego

Sieć Petriego PN może zostać zdekomponowana do zbioru łukowo rozłącznych podsieci $H = \{h_1, h_2, \dots, h_k\}$, gdzie podsieć jest opisana przez $h_r = (P_r, T_r, F_r, W_r, M_{0r})$, a każdy łuk występuje dokładnie w jednej podsieci h_i . Jest to modyfikacja definicji dekompozycji sformułowanej dla digrafów. Jednocześnie należy zaznaczyć, że w literaturze można zaobserwować stosowanie terminu dekompozycja do zbioru podsieci niebędących rozłącznymi łukowo, np. t-komponenty. W przypadkach, gdy prezentowana dekompozycja nie będzie spełniać tego warunku rozłączności łukowej, zostanie to zaznaczone w definicji.

W literaturze naukowej istnieją liczne dekompozycje, z których część została skonstruowana dla konkretnego rozszerzenia sieci Petriego, a część dla ich klasycznego wariantu.

Dekompozycje stworzone dla klasycznych sieci mogą być jednocześnie zastosowane dla większości rozszerzeń sieci Petriego. Jest to spowodowane tym, że rozszerzenia sieci t.j. TPN, nie przebudowują podstawowych zasad teorii sieci Petriego, a jedynie rozszerzają je. Najczęstszym obszarem podlegającym modyfikacjom jest reguła uruchomienia tranzycji, do której zostaje dodany dodatkowy warunek pozwalający na jej uruchomienie np. czas, czy prawdopodobieństwo uruchomienia. Dlatego metody utworzone dla klasycznego wariantu mogą zostać łatwo, często bez wprowadzania modyfikacji, efektywnie użyte dla większości rozszerzeń sieci Petriego.

Część metod dekompozycji opisuje podsieci poprzez podanie jednego zbioru wierzchołków, które stanowią rdzeń struktury, wokół którego jest budowana (indukowana) właściwa podsieć. Natomiast drugi zbiór wierzchołków, nieobecny w części definicji dekompozycji, zostaje uzyskany przez zawarcie w nim wszystkich wierzchołków połączonych łukiem z przynajmniej jednym elementem ze zbioru budującego podsieć.

Zdecydowaną większość dekompozycji uznaje jeden z typów wierzchołków za główny nośnik informacji. Tworzy to naturalny podział na podsieci, których rdzeniem jest zbiór miejsc lub zbiór tranzycji. Z tego powodu, gdy pojawi się informacja o zbiorze indukującym podsieć, oznacza to, że do podsieci należą wszystkie elementy z tego zbioru oraz wierzchołki połączone z nimi łukami.

W dalszej części rozdziału przedstawiono definicje dekompozycji do podsieci, będących istotnymi dla algorytmów porównywania sieci opracowanymi w ramach tej dysertacji.

4.2.1 t-komponent i s-komponent

Za dekompozycję do struktur odpowiadających podsieciom powstałym na bazie wsparć minimalnych t-, oraz p-niezmienników (w literaturze można znaleźć także zapis s-niezmienników) [33, 17, 15], odpowiadają t- i s-komponenty. Należy zaznaczyć, że mogą one reprezentować nierozłączne łukowo podsieci, w przeciwieństwie do pozostałych przypadków dekompozycji opisywanych w tym rozdziale.

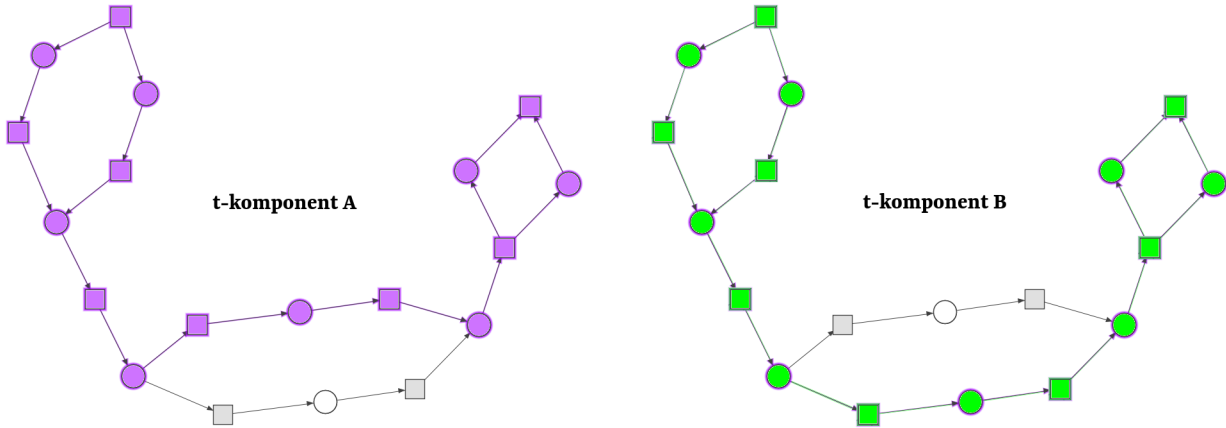
t-komponent jest to podsieć $h_r^{tc} = (P_r^{tc}, T_r^{tc}, F_r^{tc}, W_r^{tc}, M_{0r}^{tc})$ zbudowana na bazie wsparcia dla t-niezmiennika x [33] (dwie przykładowe podsieci przedstawiono na Rysunku 4.1).

$$T_r^{tc} = s(x) \quad (4.1)$$

gdzie zbiór tranzycji T_r^{tc} odpowiada zbiorowi wsparcia t-niezmiennika $s(x)$.

$$P_r^{tc} = \bullet T_r^{tc} \cup T_r^{tc} \bullet \quad (4.2)$$

gdzie zbiór P_r^{tc} zawiera wszystkie miejsca połączone łukami z tranzycjami ze zbioru T_r^{tc} .



Rysunek 4.1: Przykład dwóch t-komponentów odpowiadających dwóm t-niezmiennikom występującym w powyższej sieci.

s-komponent jest to podsieć $h_r^{sc} = (P_r^{sc}, T_r^{sc}, F_r^{sc}, W_r^{sc}, M_{0r}^{sc})$ zbudowana na bazie wsparcia dla p-niezmiennika y [15].

$$P_r^{sc} = s(y) \quad (4.3)$$

gdzie zbiór miejsc P_r^{sc} odpowiada zbiorowi wsparcia p-niezmiennika $s(y)$.

$$T_r^{sc} = \bullet P_r^{sc} \cup P_r^{sc} \bullet \quad (4.4)$$

gdzie zbiór T_r^{sc} zawiera wszystkie tranzycje połączone łukami z miejscami ze zbioru P_r^{sc} .

4.2.2 Zbiory MCT

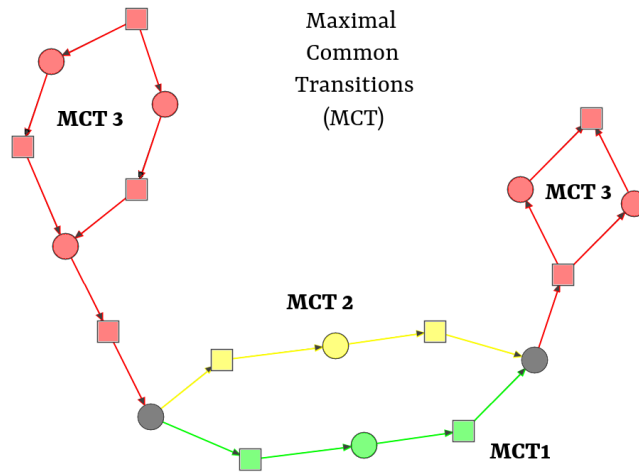
Zbiory MCT (ang. *Maximal common transition*) zostały pierwszy raz opisane w artykule [108]. Zbiór MCT jest to zbiór c , który zawiera tranzycje, będące elementami we wsparciach dokładnie tych samych t-niezmienników. Podsieci indukowane przez zbiory MCT są definiowane jako $h^M = (P_r^M, T_r^M, F_r^M, W_r^M, M_{0r}^M)$, gdzie:

$$T_r^M = c_r \quad (4.5)$$

gdzie T_r^M odpowiada pojedynczemu zbiorowi MCT c_r .

$$P_r^M = \bullet T_r^M \cup T_r^M \bullet \quad (4.6)$$

$$\forall_{p \in P_r^M} \exists_{t \in T_r^M} p \in \bullet t \vee p \in t \bullet \quad (4.7)$$



Rysunek 4.2: Przedstawiona sieć podlega dekompozycji do trzech podsieci indukowanych przez zbiory MCT. Spośród nich MCT3 reprezentuje podsieć niespójną.

Każde miejsce p ze zbioru P_r^M jest połączone łukiem z przynajmniej jedną tranzycją ze zbioru T_r^M . Podsieć indukowana przez zbiór MCT jest określana przez h^m (przykład dekompozycji został przedstawiony na Rysunku 4.2). Tranzycje, które nie pojawiają się w żadnym wsparciu t-nieziennika, tworzą nowy zbiór MCT [108].

4.2.3 Zbiory ADT

W artykule [51] z 2009 roku zostały zdefiniowane zbiory ADT (ang. Abstract Dependent Transitions). Odpowiadają one zdefiniowanym wcześniej zbiorom MCT, ale wprowadzają kilka dodatkowych elementów do teorii, istotnych z perspektywy algorytmów prezentowanych w tej pracy.

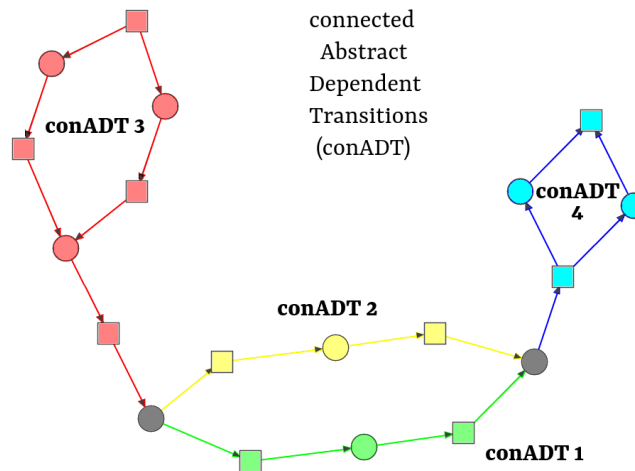
Pierwszą różnicą pomiędzy definicjami ADT i MCT, jest zbiór miejsc granicznych P_{IF} (ang. interface places set), zdefiniowany początkowo dla ADT. Jest to zbiór miejsc, które istnieją w co najmniej dwóch podsieciach h_i^M i h_j^M , co jest reprezentowane za pomocą następującego równania:

$$P_{ij}^{IF} = \bigcup_{t_i \in T_i^M, t_j \in T_j^M; i, j=1, 2, \dots, k; i \neq j} (\bullet t_i \cup t_i^\bullet) \cap (\bullet t_j \cup t_j^\bullet) \quad (4.8)$$

gdzie T_i^M jest zbiorem tranzycji podsieci h_i^M , a T_j^M podsieci h_j^M .

Drugą różnicą jest wyszczególnienie na poziomie definicji ADT spójnego wariantu indukowanej podsieci. Zbiór, na bazie którego budowana jest taka podsieć, nazywamy *spójnym zbiorem ADT* (ang. Connected ADT – conADT). Pomiedzy dwoma dowolnymi wierzchołkami w tej podsieci istnieje nieskierowana ścieżka, co jest warunkiem spójności. Przykład dekompozycji do podsieci conADT znajduje się na Rysunku 4.3.

Dla rozróżnienia pomiędzy spójnymi ADT, a ADT odpowiadającymi MCT, termin *maksymalnych zbiorów ADT* (*maxADT*) zostaje wprowadzony do określenia ADT dopuszczających niespójność indukowanych przez nie podsieci. Podsieć indukowana *conADT* jest określana za pomocą h^c , aby odróżnić ją od podsieci h^M indukowanych przez MCT i ADT.



Rysunek 4.3: Powyższa sieć zawiera cztery spójne podsieci conADT. Miejsca graniczne reprezentujące obszar styczności co najmniej dwóch podsieci zostały zaznaczone za pomocą koloru szarego.

4.2.4 t-sieć

Podsieć typu *t-sieć* reprezentuje wariant potencjalnie najmniejszych zdekomponowanych podsieci $h_r^T = (P_r^T, T_r^T, F_r^T, W_r^T, M_{0r}^T)$ [143] rozważanych w tej dysertacji. Istotną różnicą wobec poprzednich rozważanych rodzajów zdekomponowanych podsieci jest to, że w przeciwieństwie do conADT i MCT, nie wymagają one zbioru minimalnych t-nieziemienników do jej wyznaczenia.

Pojedyncza sieć t-sieć reprezentuje małą spójną podsieć indukowaną przez zbiór tranzycji połączonych przez *miejsca przepływowe*. Miejscem przepływowym nazywamy miejsce posiadające dokładnie jedną bezpośrednią tranzycję wejściową $|p^\bullet| = 1$ i jedną bezpośrednią tranzycję wyjściową $|\bullet p| = 1$.

Dwie tranzycje t_i i t_j będą częścią tej samej t-sieci, jeśli spełniają następujący warunek:

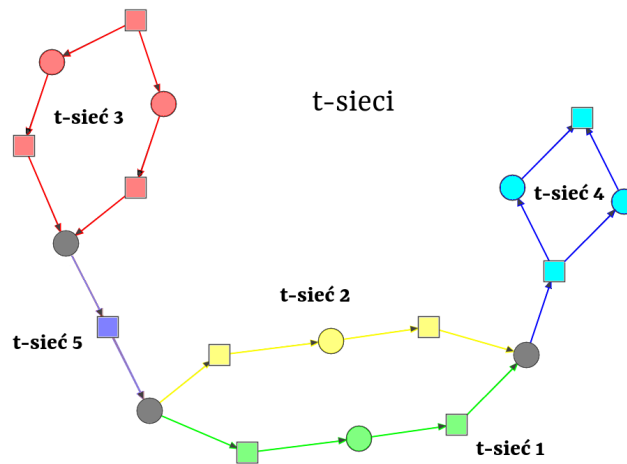
$$\forall_{t_i, t_j \in T_r} \exists_{p \in P_r} (p \in \bullet t_i \cap \bullet t_j) \wedge (|p^\bullet| = 1 \wedge |\bullet p| = 1) \quad (4.9)$$

gdzie pomiędzy każdą parą tranzycji t_i i t_j występujących w jednej podsieci h_r , musi istnieć miejsce przepływowe p .

Rezultatem bezpośredniej aplikacji dekompozycji do t-sieci na typowej sieci biologicznej będzie zbiór licznych trywialnych podsieci. Spowodowane jest to tym, że ten typ dekompozycji został zaproponowany do przeprowadzania na zredukowanych sieciach Petriego, podczas gdy modele systemów biologicznych zazwyczaj wykazują znaczną redukowalność. Z tego powodu konieczna była modyfikacja prezentowanej definicji t-sieci do postaci, która dekomponując niezredukowaną sieć biologiczną zwróci podsieci zachowujące własności t-sieci i które po redukcji odpowiadałyby podsieciom z oryginalnej definicji t-sieci. Taką podsieć nazywamy *rozszerzoną t-siecią* (ang. extended t-net) i do jej wyznaczenia używany jest następujący warunek:

$$\forall_{t_i, t_j \in T_r} \exists_{(t_i, p_l, \dots, p_k, t_j)} \forall_{p \in \{t_i, p_l, \dots, p_k, t_j\}} (|p^\bullet| = 1 \wedge |\bullet p| = 1) \quad (4.10)$$

pomiędzy każdą parą tranzycji t_i i t_j występujących w jednej podsieci T_r , istnieje nieskierowana ścieżka $\{t_i, p_l, \dots, p_k, t_j\}$, gdzie każde z zawierających się w niej miejsc jest miejscem przepływowym lub miejscem granicznym (przykład dekompozycji na Rysunku 4.4).



Rysunek 4.4: Powyżej przedstawiono dekompozycję do rozszerzonych t-sieci. W rezultacie powstało pięć małych podsieci, z których t-sieć 5 odpowiada trywialnej podsieci. W przeciwieństwie do zbiorów conADT z Rysunku 4.3, każde z miejsc granicznych zaznaczonych kolorem szarym na sieci, jest jednocześnie miejscem rozgałęziającym.

4.2.5 Zbiory MCP

Zbiór MCP jest to zbiór c , który zawiera miejsca, będące elementami we wsparciach dokładnie tych samych p-niezmienników. Podsieć $h^{MP} = (P_r^{MP}, T_r^{MP}, F_r^{MP}, W_r^{MP}, M_{0r}^{MP})$ odpowiada podsieci indukowanej przez zbiór MCP, gdzie:

$$P_r^{MP} = c_r \quad (4.11)$$

gdzie P_r^{MP} pojedynczemu zbiorowi MCP c_r .

$$T_r^{MP} = \bullet P_r^{MP} \cup P_r^{MP} \bullet \quad (4.12)$$

$$\forall_{t \in T_r^{MP}} \exists_{p \in P_r^{MP}} t \in \bullet p \vee t \in p \bullet \quad (4.13)$$

Każda tranzycja t ze zbioru T_r^{MP} jest połączona łukiem z przynajmniej jednym miejscem ze zbioru P_r^{MP} . Podsieć indukowana przez zbiór MCP jest określana przez h^{MP} . Miejsca, które nie pojawiają się w żadnym wsparciu p-niezmiennika, tworzą nowy zbiór MCP.

4.2.6 Spójne zbiory ADP

W artykule [51] oprócz definicji zbiorów ADT, pojawia się wzmianka o ich odpowiednikach bazujących na miejscach zbiorach ADP (ang. Abstract Dependent Places). Odpowiadają one zbiorom MCP, oraz posiadają swój wariant indukujący spójne podsieci conADP. Podsieć indukowana *conADP* jest określana za pomocą h^{cp} .

4.2.7 s-sieć

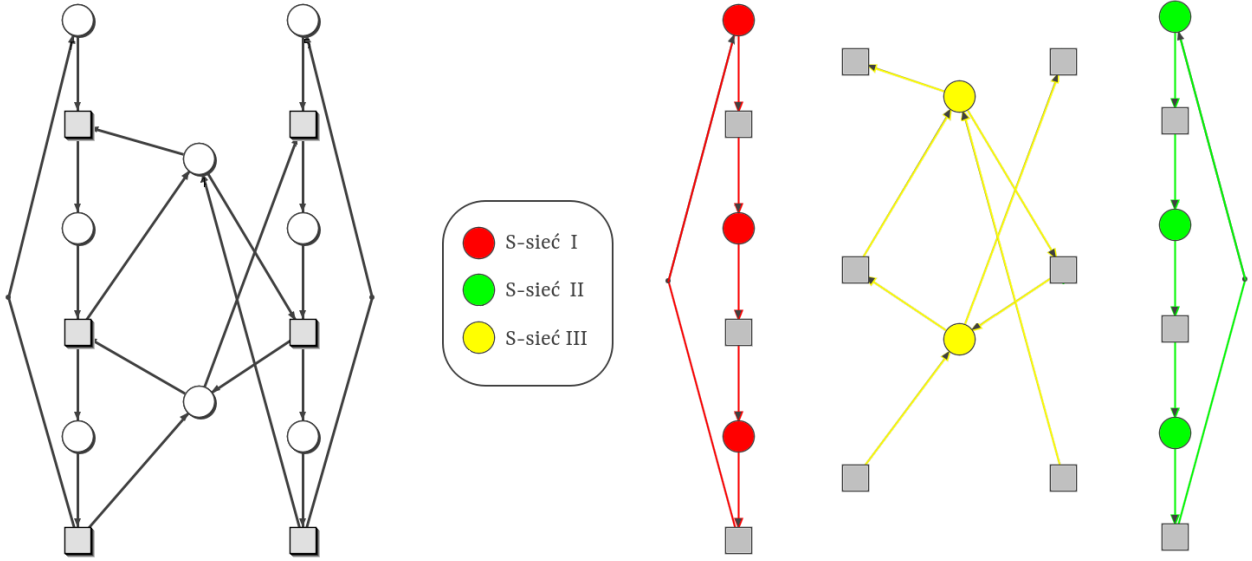
Reprezentuje typ zdekomponowanej podsieci $h_r^S = (P_r^S, T_r^S, F_r^S, W_r^S, M_{0r}^S)$ [143], zbudowany na bazie zbioru miejsc. Podobnie jak w przypadku t-sieci nie jest on zależny od p-niezmienników i posiada te same powiązania z conADP i MCP. Podsieć conADP zawiera w sobie przynajmniej jedną podsieć s-net, a podsieć indukowana MCP zawiera w sobie przynajmniej jedną podsieć conADP. Pojedyncza podsieć s-sieć reprezentuje małą spójną podsieć indukowaną przez zbiór miejsc połączonych przez *tranzycje przepływowe*. Tranzycją przepływową nazywamy tranzycję posiadającą dokładnie jedno bezpośrednie miejsce wejściowe $|t^\bullet| = 1$ i jedno bezpośrednie miejsce wyjściowe $|\bullet t| = 1$.

Dwa miejsca p_i i p_j będą częścią tej samej s-sieci, jeśli spełniają następujący warunek:

$$\forall_{p_i, p_j \in P_r} \exists_{t \in T_r} (t \in \bullet p_i^\bullet \cap \bullet p_j^\bullet) \wedge (|t^\bullet| = 1 \wedge |\bullet t| = 1), \quad (4.14)$$

gdzie pomiędzy każdą parą miejsc p_i i p_j występujących w jednej podsieci h_r , musi istnieć tranzycja przepływowa t .

Analogicznie do t-sieci, opracowany został wariant s-sieci operujący na niezredukowanych sieciach Petriego nazywany rozszerzonymi s-sieciami (es-sieciami). Do wyznaczenia takiej podsieci konieczny



Rysunek 4.5: Powyżej przedstawiono dekompozycję prostego modelu do trzech s-sieci. Każda z tranzycji występujących w tym przykładzie jest jednocześnie tranzycją graniczną dla dwóch s-sieci.

jest zbiór miejsc spełniających poniższy warunek:

$$\forall_{p_i, p_j \in P_r} \exists_{(p_i, t_l, \dots, t_k, p_j)} \forall_{t \in (p_i, t_l, \dots, t_k, p_j)} (|t^\bullet| = 1 \wedge |\bullet t| = 1) \quad (4.15)$$

poniędzy każdą parą miejsc p_i i p_j występujących w jednej podsieci P_r , istnieje nieskierowana ścieżka $\{p_i, t_l, \dots, t_k, p_j\}$, gdzie każda z zawierających się w niej tranzycji jest tranzycją przepływową lub tranzycją graniczną.

4.2.8 P1-sieć / komponent maszyny stanowej

Sieć Petriego nazywamy P1-siecią, jeśli można ją zdekomponować do podsieci $h_r^{P1} = (P_r^{P1}, T_r^{P1}, F_r^{P1}, M_0^{P1})$, $r = 1, \dots, |H^{P1}|$, które odpowiadają łukowo rozłącznym s-komponentom. Ponadto, każda z podsieci posiada dokładnie 1 token we wszystkich stanach osiągalnych dla niej [86, 87, 88].

$$P_r^{P1} = s(y_z) \quad (4.16)$$

$$T_r^{P1} = \bullet P_r^{P1} \cup P_r^{P1} \bullet \quad (4.17)$$

$$\sum_{i=1}^{|P_r^{P1}|} M_0^{P1}(i) = 1 \quad (4.18)$$

gdzie $M_0^{P1}(i)$ jest liczbą tokenów znajdujących się w miejscu p_i . Dla każdego stanu M_e^{P1} osiągalnego w podsieci, ta zależność zostaje zachowana, tj.:

$$\sum_{i=1}^{|P_r^{P1}|} M_e^{P1}(i) = 1 \quad (4.19)$$

P1-sieci były wykorzystywane w modelach biologicznych reprezentujących system immunologiczny człowieka [86, 87, 88]. W literaturze można znaleźć odniesienia do identycznych podsieci nazywanych

komponentami maszyny stanowej (ang. State Machine Component z akronimem SMC) i używanych do zastosowań poza biologicznych np. budowa kontrolerów [129, 131].

4.2.9 Podsieci funkcyjne

Funkcyjna podsieć $h_r^Z = (P_r^Z, T_r^Z, F_r^Z, M_0^Z)$ spełnia następujące własności:

$$\bullet T_r^Z \cap T_r^Z \bullet = \emptyset \quad (4.20)$$

$$\forall_{p \in \bullet T_r^Z} \bullet p = \emptyset \quad (4.21)$$

$$\forall_{p \in T_r^Z \bullet} p \bullet = \emptyset \quad (4.22)$$

$$F_r^Z \subseteq (P_r^Z \times T_r^Z) \cup (T_r^Z \times P_r^Z) \quad (4.23)$$

Funkcyjna podsieć h_r^Z jest podsiecią PN , kiedy każde miejsce p z podsieci h_r^Z jest bezpośrednim miejscem wejściowym albo wyjściowym, ale nigdy oboma:

$$\forall_{p \in \bullet P_r^Z} \nexists_{t \in T_r^Z} t \in \bullet p \quad (4.24)$$

$$\forall_{p \in P_r^Z \bullet} \nexists_{t \in T_r^Z} t \in p \bullet \quad (4.25)$$

$$\forall_{p \in Q} \nexists_{t \in T_r^Z} t \in p \bullet \wedge t \in \bullet pv \quad (4.26)$$

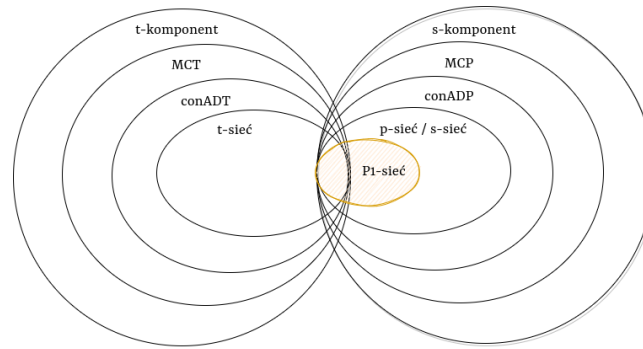
gdzie $Q = P_r^Z \bullet \cup \bullet P_r^Z$ odpowiada zbiorowi miejsc wewnętrznych.

Podsieci funkcyjne zostały zaproponowane w celu analizy i weryfikacji złożonych protokołów telekomunikacyjnych.

4.3 Dekompozycje bazujące na fragmentach t-komponentów

Niezmienniki obu typów są powszechnie używane przy analizowaniu modeli opartych o sieci Petriego. W niektórych publikacjach można spotkać potoczne uproszczenia łączące bezpośrednio wektor, reprezentujący niezmiennik, ze strukturą indukowaną przez wsparcie tego wektora. Podsieci takie zostały już zdefiniowane w połowie lat 80-tych, jednak terminologia t- i s-komponentów nie rozpowszechniła się w obszarze zastosowań bioinformatycznych.

Z metod dekompozycji prezentowanych w tej pracy t- i s-komponenty reprezentują największe, a przez to najbardziej skomplikowane podsieci. To, w połączeniu z potencjalnie wysoką liczbą niezmienników występujących w analizowanych modelach, może stanowić ich największą zaletę, jak i wadę. Podczas wyznaczania zbioru minimalnych niezmienników może wystąpić wykładniczy wzrost liczby cząstkowych wektorów koniecznych do wyznaczenia tego zbioru [67]. Alternatywą jest przeprowadzenie analiz na podsięciach opartych o fragmenty t- i s-komponentów.



Rysunek 4.6: Diagram przedstawiający relacje zawierania się struktur występujących jednocześnie w podsieciach z kilku dekompozycji

4.3.1 Rodzina MCT, conADT i t-sieć

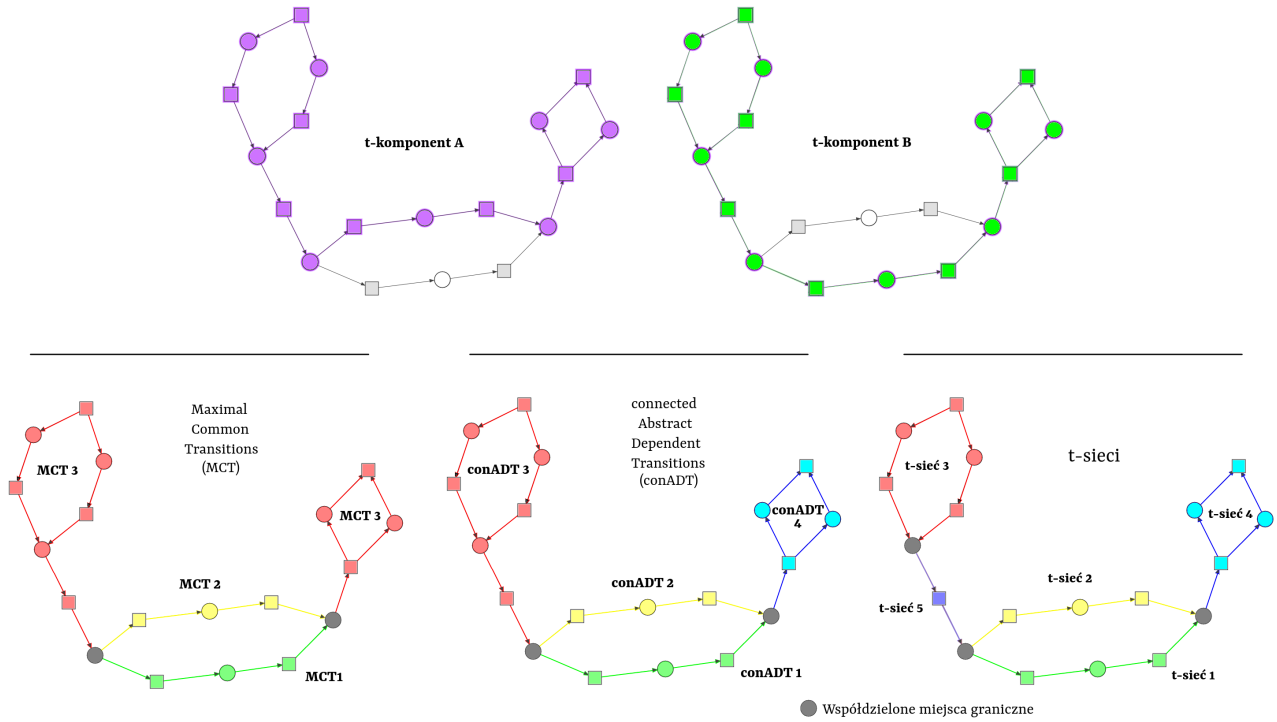
Struktury bazujące na zbiorach MCT, spójne podsieci conADT i t-sieć ze względu na silne podobieństwa i powiązania przedstawione zostaną razem w tej sekcji. Dwa pierwsze z nich powstały z zamiarem zastosowaniem do analizy sieci biologicznych. Struktury ADT powstały jako rozszerzenie koncepcji MCT [108]. Natomiast dekompozycja do postaci t-sieci powstała niezależnie od nich, jako czysto matematyczna metoda analizy dla sieci Petriego. Przykład dekompozycji do danych wariantów przedstawiono na Rysunku 4.7.

Relacje pomiędzy strukturami uzyskiwanymi za pomocą wymienionych dekompozycji można określić jako przejście od ogólnych podsieci MCT/maxADT, przez conADT, do najmniejszej formy t-sieci (przykład na Rysunku 4.7). Podsieć MCT zawiera w sobie przynajmniej jeden komponent o postaci conADT, podczas gdy conADT posiada w sobie przynajmniej jeden komponent o postaci t-sieci. Istotną zaletą ostatniej dekompozycji jest brak konieczności wyznaczania niezmienników do przeprowadzenia analizy.

Struktury oparte o MCT mogą tworzyć niespójne podsieci. Każda spójna podsieć będzie odpowiadać strukturze conADT. Natomiast różnice pomiędzy conADT a t-siecią polegają na ich relacji z miejscami rozgałęziającymi, co dokładnie opisano w Rozdziałach 5 i 9.

Elementem łączącym wymienione typy podsieci jest rola jaką pełnią w nich tranzycje, będące ich rdzeniem. Tworzą one rodzinę dekompozycji dzielącą sieć na coraz to mniejsze podsieci, reprezentując relację od ogółu (sieci indukowane przez MCT), do szczegółu (t-sieci).

Autor publikacji [51] twierdzi, że "Z biologicznej perspektywy conADT reprezentują najmniejszą strukturę o biologicznym znaczeniu" [51]. Jak wobec tego interpretować struktury t-sieci, reprezentujące zazwyczaj podsieci izomorficzne do podsieci opartych o conADT lub mniejsze? Analiza podobieństwa t-sieci z conADT i MCT wykazała, że t-sieci mogą opisywać struktury interesujące biologicznie, będąc jednocześnie jednym z kilku elementów składowych conADT. Z punktu widzenia algorytmicznego i pesymistycznych przypadków, struktury oparte o t-sieci są preferowane nad conADT oraz MCT, ze względu na niską czasową złożoność obliczeniową i brak wcześniejszej potrzeby wyznaczenia t-niezmienników.



Rysunek 4.7: Rysunek przedstawia relacje pomiędzy dekompozycjami indukowanymi przez zbiory tranzycji MCT, lub conADT. Najmniejszą częścią składową jest reprezentowana przez podsieć rozszerzoną t-sieć, reprezentowaną osobnym kolorem. Podsieci bazujące na conADT mogą składać się z kilku podsieci t-sieci, podczas gdy podsieci bazujące na MCT z kilku podsieci conADT i t-sieci.

4.4 Dekompozycje bazujące na fragmentach p-niezmienników

4.4.1 Rodzina MCP, conADP i s-sieć

Pozorna lustrzaność

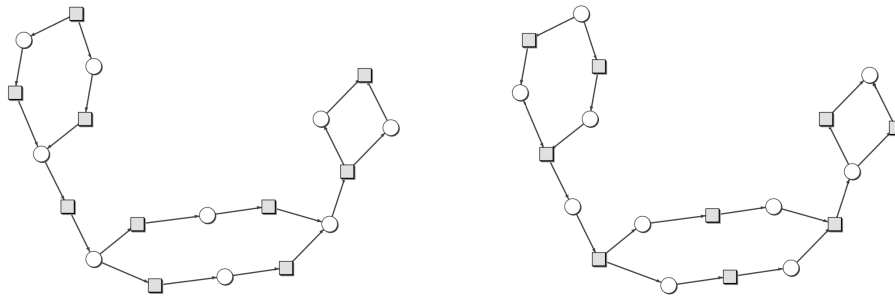
Proces wyznaczania zbioru minimalnych p-niezmienników jest taki sam jak dla t-niezmienników [67]. Algorytm pozostaje bez zmian, natomiast różnicą są dane wejściowe. Macierz incydencji jest transponowana, aby pożądaný typ wierzchołka odpowiadał za wiersze.

Wszystkie dekompozycje bazujące na t-niezmiennikach, jak np. zbiory MCT, ADT, t-sieci, można zaadoptować dla p-niezmienników w postaci MCP, ADP, s-sieci. Wyznaczające je algorytmy nie muszą zostać poddane znaczącym zmianom, a jedynie dostosowane do operowania na nowym zbiorze danych wejściowych. Przykładem jest wspomniane wcześniej wyznaczanie zbioru minimalnych p-niezmienników. Problemem pozostaje jednak interpretacja wyników. Mimo wspomnianych podobieństw, miejsca i tranzycje pełnią różne funkcje w kontroli przepływu tokenów w sieciach Petriego. Efektem jest to, że podprocesy opisane przez dekompozycje MCP, ADP i s-sieci, różnią się znacząco od podprocesów opisywanych przez MCT, ADT, t-sieci. Należy zauważyć, że nowe dekompozycje bazujące na p-niezmiennikach nie biorą pod uwagę rozkładu tokenów w podsieciach.

Przeprowadzanie analiz bazujących na miejscach bez uwzględniania informacji o zawartych w nich tokenach, czy odniesienia do stanu całej sieci, pozbawia kluczowych z perspektywy p-niezmienników informacji. Aby uzyskać efektywne metody, mające powszechne zastosowanie w teorii sieci Petriego, potrzebne są modyfikacje tych dekompozycji, czy to w postaci częściowego uwzględnienia stanu (sieci P1), lub głębszych zmian w definicjach podsieci. Mimo równoległego zaproponowania dla miejsc odpo-

wiedników wspomnianych powyżej dekompozycji nie zaobserwowano zainteresowania nimi. Podobna sytuacja występuje w przenoszeniu analiz stworzonych na bazie p-niezmienników. Nie oznacza to, że uzyskane podsieci są bezwartościowe, jednak do czasu udowodnienia przedstawianych przez nie właściwości, lub opracowania metod analizy bazujących na MCP, ADP, s-sieciach, ich potencjał pozostaje niezbadany.

Wszystkie trzy dekompozycje MCP, conADP, s-sieć, zachowują te same relacje między sobą co ich odpowiedniki oparte na tranzycjach MCT, conADP i t-sieć. Widać to na podstawie dualnych sieci Petriego opisanych przez T. Muratę w [83]. Sieć dualną dla sieci PN_A nazywamy sieć PN_B , w której typy wierzchołków zostały zamienione na przeciwnego typu (Rysunek 4.8).



Rysunek 4.8: Powyżej przedstawiono dwie sieci reprezentujące relację dualności. Miejscom w sieci po lewej stronie odpowiadają tranzycje w sieci po prawej stronie, tak jak tranzycjom po lewej stronie odpowiadają miejsca po prawej stronie. Wynikiem tej dualności jest to, że 2 t-niezmiennikom z lewej sieci odpowiadają 2 p-niezmienniki w sieci prawej.

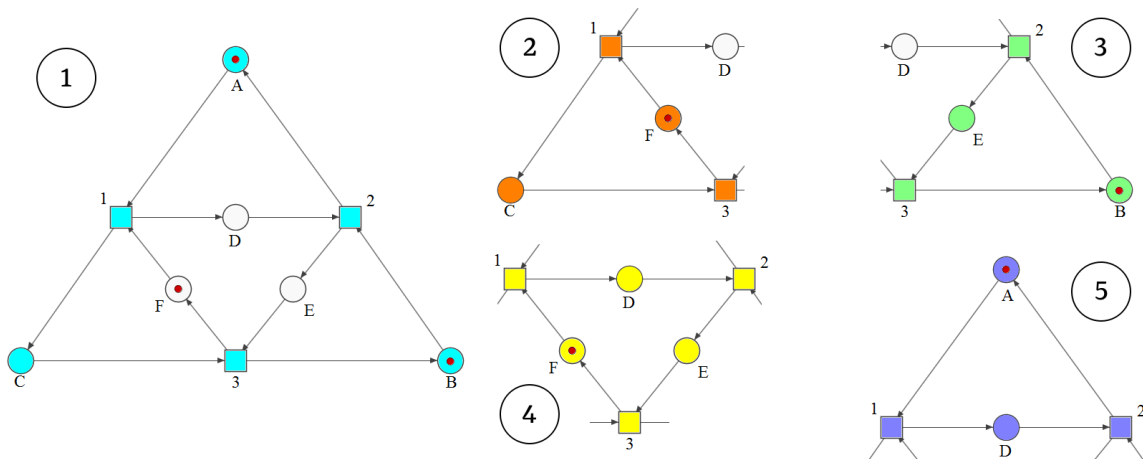
Mimo przedstawionych powyżej problemów w interpretacji tych struktur mają one potencjał z punktu porównywania sieci Petriego. Podczas wyznaczania niezmienników może dojść do wykładniczego wzrostu liczby cząstkowych wektorów (problem opisany w [67]), w efekcie blokując możliwość wyznaczenia podsieci opartych o niezmienniki takich jak MCT czy conADT. Jednak pojawienie się problemu wykładniczego wzrostu liczby cząstkowych wektorów przy wyznaczaniu t-niezmienników nie oznacza, że problem ten wystąpi przy wyznaczaniu p-niezmienników dla tej samej sieci. Liczba t-niezmienników jest niezależna od liczby p-niezmienników występujących w tej samej sieci. Dzięki temu istnieje możliwość porównania strukturalnego sieci używając MCP, ADP czy s-sieci. Uzyskujemy jednak tylko informacje o podobieństwie struktur i jak zostało wspomniane wcześniej, do czasu opracowania odpowiedniej interpretacji (interpretacja biologiczna struktur bazujących na s-sieci dotychczas nie pojawiła się w literaturze), zostają one ograniczone tylko do podobieństwa strukturalnego. Jednocześnie należy pamiętać, że istnieją sieci, w których problem wykładniczego wzrostu liczby cząstkowych wektorów występuje przy wyznaczaniu obu typów niezmienników.

4.4.2 P1-sieć

W przeciwieństwie do metod opisanych powyżej podsieci P1-sieci zostały opracowane od początku bazując na p-niezmiennikach. Początkowo były stosowane do analiz maszyn stanowych reprezentujących procesy przemysłowe [131, 129], a dopiero później pojawiła się propozycja ich zastosowania w sieciach biologicznych [86]. Od modelu wymagane jest, aby przed rozpoczęciem dekompozycji spełniał własności żywotności, konserwatywności i bezpieczeństwa [131]. Sieć nazywana jest żywotną (ang. *liveness*), jeśli w każdym stanie osiągalnym w sieci, dla każdej z tranzycji t istnieje pewna sekwencja

uruchomień tranzycji, która ją uruchomi. Sieć nazywana jest konserwatywną (ang. *conservative*) jeśli każda z tranzycji w momencie uruchomienia pobiera tyle samo tokenów z miejsc bezpośrednio wejściowych, co generuje tokenów do miejsc bezpośrednio wyjściowych. Sieć nazywamy bezpieczną (ang. *safe*), jeśli w każdym z jej miejsc liczba tokenów nigdy nie przekroczy 1.

Podsiec P1-siec jest jednocześnie s-komponentem, ale nie każdy s-komponent jest P1-siecią. W miejscach należących do podsieci P1-sieci znajduje się tylko jeden token i w wyniku uruchomień tranzycji jego liczba tokenów nie zmienia się. Jest to zagwarantowane przez spełnienie własności bezpiecznych sieci Petriego, którą musi spełniać oryginalna sieć przed dekompozycją. Sieć którą można zdekomponować do podsieci P1-sieci nazywamy P1. Jeśli model sieci Petriego jest interpretowany jako maszyna stanowa, P1-siec odpowiada jej pojedynczemu komponentowi (Rysunek 4.9).



Rysunek 4.9: Przykład sieci Petriego zawierającej pięć podsieci indukowanych przez p-niezmienniki. Z nich cztery podsieci (przypadki od 2 do 5) odpowiadają poprawnym P1-sieciom. W przypadku 1, na modelu niebieskim kolorem zaznaczono podsiec indukowaną p-niezmiennikiem, która nie jest P1-siecią ponieważ zawiera dwa miejsca przechowujące token w tym samym stanie sieci.

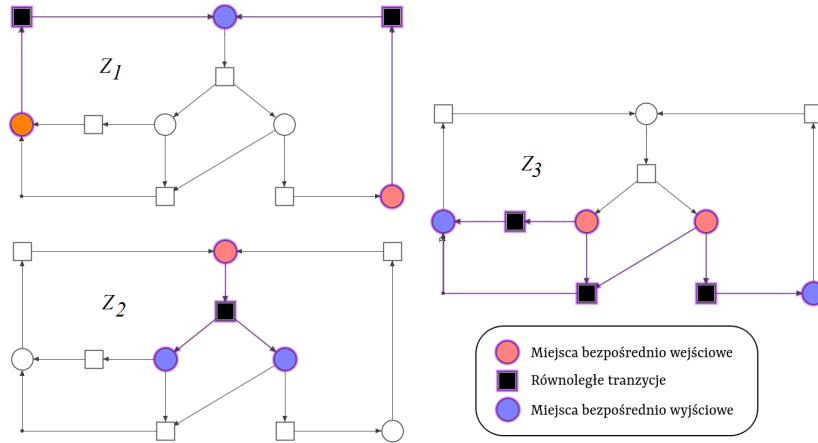
Dekompozycja ta jest ograniczona do modeli pokrytych przez rozłączne minimalne p-niezmienniki, gdzie wsparcie każdego z nich zawiera dokładnie jeden token (Rysunek 4.9). Ogranicza to zastosowanie tej dekompozycji tylko do modeli spełniających wspomniane warunki. Jednak mimo to obserwowane są jej zastosowania przy programowaniu mikrokontrolerów, maszyn stanowych, oraz modelowaniu systemu odpornościowego. Ten ostatni przykład stanowi bazę kilku artykułów naukowych [86, 87, 88] podejmujących analizy sieci o wysokim stopniu wewnętrznych powiązań, przez badanie sekwencji uruchomień tranzycji zachodzących w podsieciach.

W porównaniu z innymi metodami dekompozycji przedstawionymi w tym rozdziale, przy konstruowaniu tych podsieci analizie podlega rozkład tokenów, nieobecny w metodach bazujących na t-niezmiennikach.

4.5 Podsieci funkcyjne

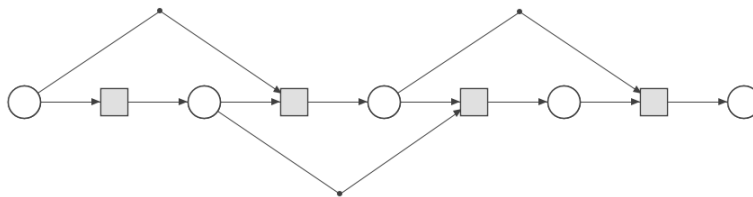
Osobną kategorię stanowią sieci funkcyjne opracowane przez Zeitseva w 2004 roku [137]. W odróżnieniu od poprzednich wymienionych dekompozycji powstały one w wyniku pracy nad analizą czasowych sieciach Petriego [139]. Jednocześnie ani same podsieci funkcyjne ani algorytm wyznaczający je, nie używają żadnych czasowych elementów z TPN lub DPN. Pozwala to na wyznaczenie ich dla klasycz-

nych sieci Petriego oraz różnych rozszerzeń. Drugą ważną charakterystyką jest brak potrzeby wyznaczenia niezmienników jako kroku przy tworzeniu sieci funkcyjnych. Autor dekompozycji wspomina jednocześnie, że mogą one zostać użyte do ograniczenia złożoności przy wyznaczaniu niezmienników [140].



Rysunek 4.10: Przykład podziału na podsieci funkcyjne. W ramach każdej podsieci zaznaczono kolorami miejsca bezpośrednio wejściowe i bezpośrednio wyjściowe.

Podsieć funkcyjna reprezentowana jest przez zbiór tranzycji oraz wyznaczone dla niego zbiory bezpośrednich miejsc wejściowych i wyjściowych. Tranzycje występujące w jednym zbiorze współdzielą z przynajmniej jedną inną tranzycją z tego zbioru bezpośrednie miejsca wejściowe lub bezpośrednie miejsca wejściowe, przy ograniczeniu że żadne bezpośrednie miejsce wejściowe nie może być bezpośrednim miejscem wyjściowym innej tranzycji ze zbioru. Przykład dekompozycji do podsieci funkcyjnych został przedstawiony na Rysunku 4.10. Obszernym źródłem informacji na temat własności i zastosowań sieci funkcyjnych jest książka [138].



Rysunek 4.11: Przykład sieci niemożliwej do podziału na sieci funkcyjne ze względu na sekwencyjne zależności pomiędzy tranzycjami.

W przeciwieństwie do poprzednio opisanych dekompozycji, w wyniku podziału na podsieci funkcyjne mogą pojawić się fragmenty sieci, które nie reprezentują podsieci funkcyjnych. Przykład takiego nienadającego się do dekompozycji fragmentu sieci został przedstawiony na Rysunku 4.11. Fakt, że sieć może nie być w pełni pokryta przez podsieci funkcyjne, ma istotne znaczenie dla metody porównywania opisywanej w Rozdziale 9.

Rozdział 5

Wierzchołki rozgałęziające

Relacje pomiędzy dwoma wierzchołkami połączonymi (przynajmniej jedną) krawędzią lub łukiem stanowią bazę dla analizy modeli przedstawionych za pomocą grafów i sieci. Jednak w konkretnych sytuacjach podczas ich analizy istotna informacja może się znajdować nie w sąsiednim wierzchołku a w którymś z jego następników. W niniejszej rozprawie zaproponowano dodanie do teorii sieci Petriego formalnej definicji dla wierzchołków rozgałęziających (ang. *branching vertices*) i ich relacji z sąsiadującymi wierzchołkami rozgałęziającymi, nazywanymi ich końcówkami (ang. *endpoints*), będącymi kluczowym elementem metod analizy modeli opartych o sieci Petriego. Definicje opisane w tym rozdziale będą stanowić podstawę dla metod opisanych w Rozdziałach 8 i 9

5.1 Definicje

Wierzchołkiem rozgałęziającym v^{br} nazywamy taki wierzchołek, dla którego liczba łuków wchodzących lub łuków wychodzących jest większa niż 1. W zależności od typu wierzchołka rozróżniamy zbiór tranzycji rozgałęziających $T^{br} \subseteq T$:

$$T^{br} = \{t_1^{br}, \dots, t_n^{br}\} : |t^{br\bullet}| > 1 \vee |\bullet t^{br}| > 1 \quad (5.1)$$

oraz zbiorów miejsc rozgałęziających $P^{br} \subseteq P$:

$$P^{br} = \{p_1^{br}, \dots, p_n^{br}\} : |p^{br\bullet}| > 1 \vee |\bullet p^{br}| > 1 \quad (5.2)$$

Na zbiór wierzchołków rozgałęziających V^{br} składa się suma zbiorów T^{br} i P^{br} :

$$V^{br} = T^{br} \cup P^{br} \quad (5.3)$$

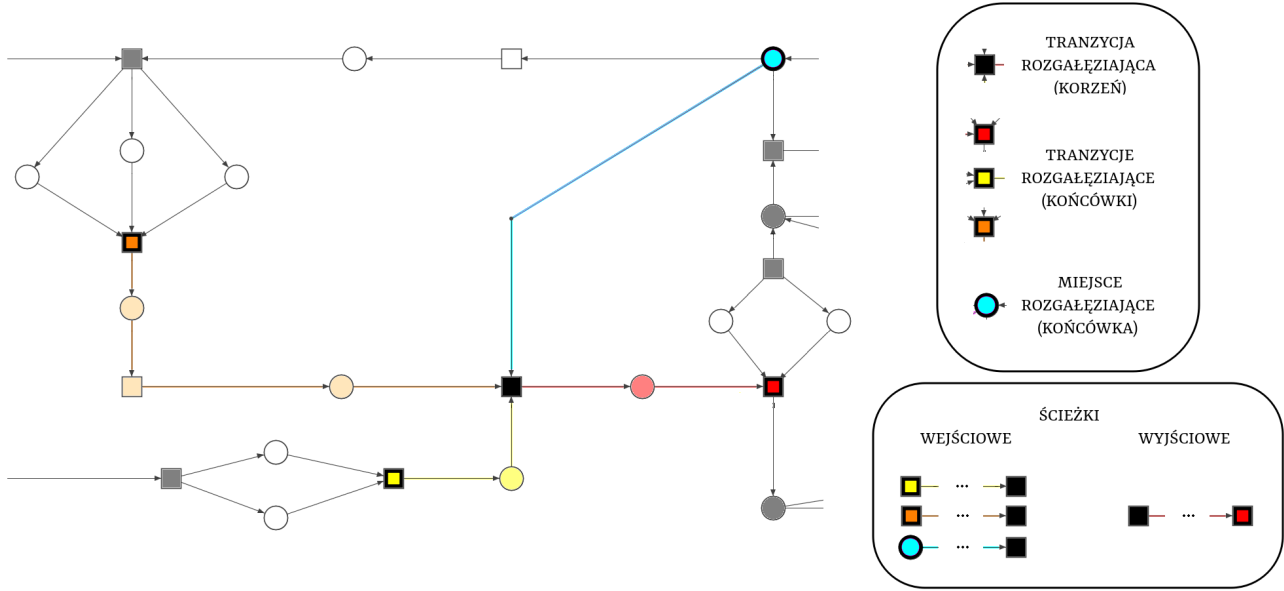
co może zostać zapisane w analogiczny sposób do wzorów (5.1) i (5.2):

$$V^{br} = \{v_1^{br}, \dots, v_n^{br}\} : |v^{br\bullet}| > 1 \vee |\bullet v^{br}| > 1 \quad (5.4)$$

W przypadkach, gdy opisywane własności zachodzą dla tranzycji i miejsc, stosowane jest odniesienie do zbioru wierzchołków v^{br} .

Istotną kwestią jest relacja występująca pomiędzy dwoma wierzchołkami rozgałęziającymi połączonymi skierowaną ścieżką, w której wierzchołki rozgałęziające występują tylko na pierwszej i ostatniej

pozycji. Każdy wierzchołek rozgałęziający v^{br} może zostać uznany za korzeń, a wtedy wszystkie wierzchołki rozgałęziające połączone z nim bezpośrednią ścieżką są jego końcówkami. Korzeń może posiadać samego siebie jako końcówkę. Każdy wierzchołek v^{br} posiada niepusty zbiór końcówek co przedstawiono na przykładzie wierzchołka rozgałęziającego wraz z jego czterema końcówkami na Rysunku 5.1. Na potrzebę analizy każda końcówka jest rozróżniana ze względu na jej typ i skierowanie łączącej ją ścieżki.



Rysunek 5.1: Fragment sieci Petriego przedstawia relację pomiędzy wierzchołkami rozgałęziającymi. Czarnym kolorem zaznaczono tranzycję t_0 pełniącą rolę korzenia, a pozostałymi kolorami o pogrubionych czarnych obwódkach jej końcówki ze zbioru $E = [2, 1, 1, 0]$ (wzór 5.7), dwie wejściowe tranzycje rozgałęziające (kolory pomarańczowy i żółty), jedną wyjściową (kolor czerwony) i jedno rozgałęziające miejsce wejściowe (kolor błękitny). Jaśniejszym odcieniem zaznaczono wierzchołki przepływowe występujące w ścieżce pomiędzy korzeniem a daną końcówką. Szarym oznaczono pozostałe wierzchołki rozgałęziające niebędące w relacji z t_0 . Całość interakcji tranzycji t_0 z jej końcówkami można przedstawić w postaci pary $L = (t, [2, 1, 1, 0])$.

Dla każdego wierzchołka v^{br} rozgałęziającego rozróżniamy cztery podzbiory zawierające końcówki odpowiadające kolejno wchodzącym ($T_{in}^{v^{br}}$) i wychodzącym ($T_{out}^{v^{br}}$) tranzycjom oraz wchodzącym ($P_{in}^{v^{br}}$) i wychodzącym ($P_{out}^{v^{br}}$) miejscami. Są one opisywane za pomocą funkcji $B(v^{br})$, która przyjmuje wartości przedstawiane w postaci wektora odpowiadającego końcówkom wierzchołka v^{br} :

$$B(v^{br}) = [T_{in}^{v^{br}}, T_{out}^{v^{br}}, P_{in}^{v^{br}}, P_{out}^{v^{br}}] \quad (5.5)$$

Wierzchołek v^{br} wraz ze zbiorem wierzchołków rozgałęziających będących jego końcówkami, jest opisywany za pomocą:

$$R(v^{br}) = (v^{br}, B(v^{br})) \quad (5.6)$$

Na potrzeby analizy porównawczej metod Dystansu Względnej Częstotliwości Grafiów (RGF) i opisaną w Rozdziale 8 Dystansu Względnej Częstotliwości Rozgałęziających Wierzchołków (RBF), wprowadzono funkcję $E(v^{br})$ przyjmującą wartość wektora:

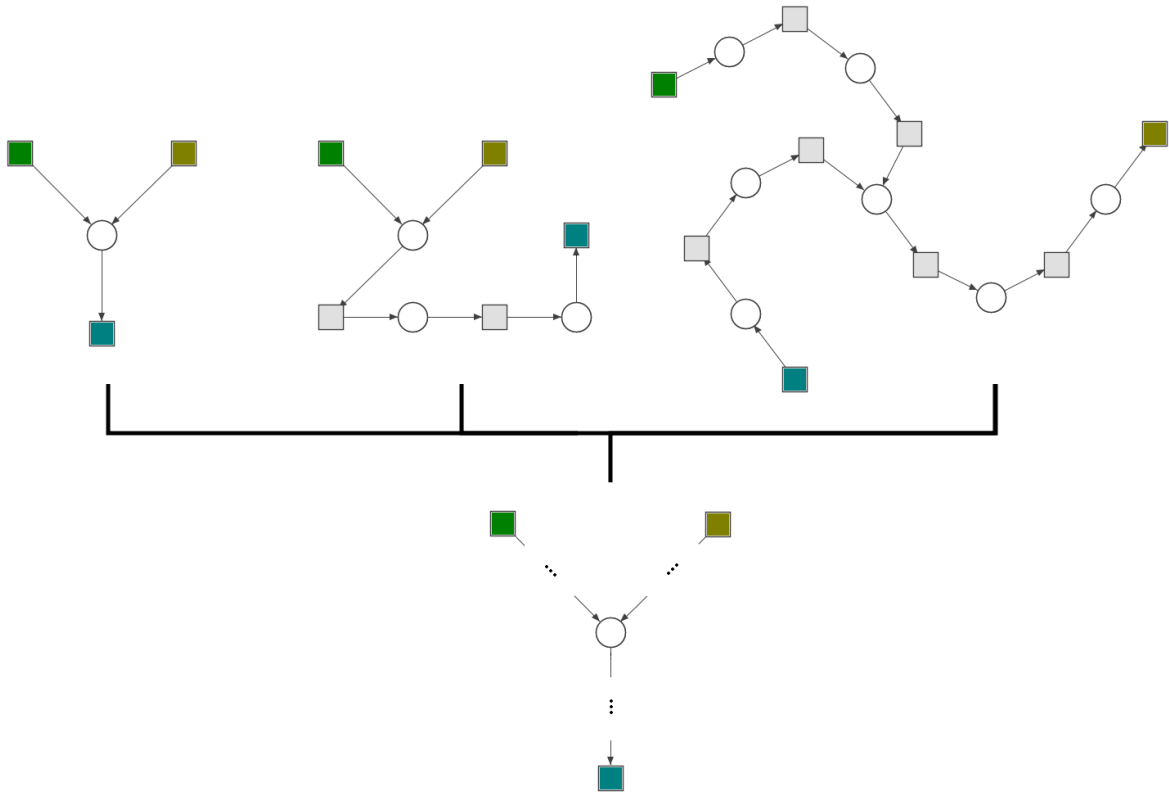
$$E(v^{br}) = [|T_{in}^{v^{br}}|, |T_{out}^{v^{br}}|, |P_{in}^{v^{br}}|, |P_{out}^{v^{br}}|], \quad (5.7)$$

gdzie kolejne pozycje w wektorze odpowiadają licznosciom podzbiorów końcówek konkretnych typów (przykład znajduje się w opisie Rysunku 5.1). Wektor $E(v^{br})$ jest przypisany do pary uporządkowanej $L(v^{br})$:

$$L(v^{br}) = (f(v^{br}), E(v^{br})) \quad (5.8)$$

gdzie funkcja $f(v^{br})$ zwraca literę T w przypadku gdy v^{br} jest tranzycją lub literę P w przypadku gdy jest miejscem. Przykładem takiego zapisu jest $(P, [2, 1, 0, 0])$ z Rysunku 5.2 przedstawiający miejsce rozgałęziające z dwoma tranzycjami wejściowymi i jedną wyjściową jako swoimi końcówkami.

Relację $R(v^{br})$ pomiędzy wierzchołkiem rozgałęziającym v^{br} , a jego końcówkami $B(v^{br})$ można przedstawić jako zbiór ścieżek, w których wierzchołek v^{br} występuje na pierwszej lub ostatniej pozycji w ścieżce (lub obu w przypadku cyklu), a drugi koniec ścieżki zajmuje jeden z elementów z podzbiorów z wektora $B(v^{br})$.



Rysunek 5.2: Przykład trzech podsieci, które reprezentowane są przez jedną strukturę $L = (P, [2, 1, 0, 0])$

Struktura $R(v^{br})$ może przedstawiać rodzinę podsieci zbudowanych wokół wierzchołka rozgałęziającego v_i^{br} pełniącego rolę punktu centralnego, połączonego ścieżkami różnej długości, ale o zadanym skierowaniu z końcówkami ze zbioru $B^{v^{br}}$. Przykładowa rodzina takich podsieci została przedstawiona na Rysunku 5.2, które po redukcji ścieżek do postaci pojedynczego łuku przedstawiałyby izomorficzne struktury.

5.2 Wierzchołki rozgałęziające a algorytmy grafowe

Wierzchołki o wyższych stopniach występujące w grafach oraz sieciach Petriego mają negatywny wpływ na efektywność niektórych algorytmów, np. algorytmu zliczania wszystkich grafletów o złożoności czasowej $O(ed^2)$, gdzie e jest liczbą krawędzi a d to najwyższy stopień wierzchołka w sieci [59]. Występujące w literaturze definicje wierzchołków wyższego stopnia i ich relacji z sąsiedztwem, jak np. interpretacja znana z bramek logicznych *OR* dla miejsc i *AND* dla tranzycji [58], miejsc granicznych dla podsieci ADT [51], tylko częściowo pokrywają się z przedstawioną definicją wierzchołków rozgałęziających. Zdarzają się też sytuacje zastosowania wierzchołków rozgałęziających bez definiowania ich w żaden sposób, np. w algorytmie wyznaczania niezmienników [67], gdzie pod koniec działania pierwszego etapu algorytmu następuje podział do podsieci ograniczonych przez miejsca rozgałęziające.

Duża część metod analitycznych opracowanych dla sieci Petriego i stosowanych w biologii systemowej bazuje na niezmiennikach, przykładowo analizy oparte o wspomniane wcześniej zbiory MCT [67] oraz klastrowanie [83]. Dlatego ważnym jest spojrzeć z perspektywy niezmienników na zachowanie wierzchołków rozgałęziających i określić relacje pomiędzy nimi.

Dwa algorytmy opracowane na potrzeby tej rozprawy i przedstawione w Rozdziałach 8 oraz 9 korzystają z wierzchołków rozgałęziających, mimo że reprezentują różne podejścia do porównywania sieci Petriego (rozkład wierzchołków/drzew i porównywanie podsieci).

5.2.1 Wierzchołki rozgałęziające a niezmienniki w sieciach Petriego

Tranzycje należące do wsparcia każdego t -niezmiennika (mowa o przypadkach prawidłowych t -niezmienników oraz sub-, sur- t -niezmienników opisanych za pomocą wzorów (2.6)-(2.8)), mogą zostać podzielone na trzy klasy:

- 1.) *trywialna tranzycja przepływowa* t posiada dokładnie jeden łuk wejściowy i jeden łuk wyjściowy.

$$|t^\bullet| = 1 \wedge |\bullet t| = 1, \quad (5.9)$$

- 2.) *tranzycja rozgałęziająca* t^{br} posiada więcej niż jeden wejściowy lub więcej niż jeden wyjściowy łuk.

$$|t^{br\bullet}| \geq 2 \vee |\bullet t^{br}| \geq 2, \quad (5.10)$$

- 3.) *trywialne tranzycje wejściowe i wyjściowe* w zależności od interpretacji, mogą być traktowane jako zwykłe tranzycje przepływowe, lub mogą być traktowane jako osobny niezależny przypadek, gdzie są interpretowane jako źródło sygnałów wejściowych dochodzących do modelowanego systemu z jego otoczenia, oraz jako wyjście sygnałów z systemu do otoczenia.

Trywialna tranzycja wejściowa t jest opisana wzorem: ,

$$|t^\bullet| = 1 \wedge |\bullet t| = 0. \quad (5.11)$$

Trywialna tranzycja wyjściowa t jest opisana wzorem:

$$|t^\bullet| = 0 \wedge |\bullet t| = 1 \quad (5.12)$$

Należy zaznaczyć, że tranzycje rozgałęziające mogą pełnić rolę tranzycji wyjściowej, lub wejściowej. Przykładowa rozgałęziająca tranzycja wyjściowa będzie tranzycją opisaną wektorem $[2, 0, 0, 0]$ dla funkcji E , a przykładem rozgałęziającej tranzycji wejściowej będzie tranzycja opisana wektorem $E = [0, 3, 0, 0]$ dla funkcji E .

Przepływowe tranzycje reprezentują najprostszy pojedynczy krok w procesie modelowanym za pomocą t-komponentu w sieci Petriego. W przypadkach tranzycji wejściowych i wyjściowych, będzie to pierwszy lub ostatni krok w t-komponencie. Tranzycje przepływowe podlegają redukcjom opisanym w podrozdziale 2.2.3.

Tranzycje rozgałęziające są kluczowym elementem t-komponentów o nietrywialnej strukturze. Z perspektywy teorii sieci Petriego są one fundamentem dla relacji współbieżności (ang. concurrency). Oznacza ona, że w sieci znajdują się przynajmniej dwie tranzycje nie współdzielące miejsc bezpośrednio wejściowych lub wyjściowych, które mogą zostać uruchomione jednocześnie [58]. W najprostszym przypadku odpowiada to dwóm równoległym łańcuchom tranzycji i miejsc, zaczynających się i kończących na tranzycjach rozgałęziających.

Podsieci indukowane przez t-niezmienniki można podzielić na dwa typy struktur. Pierwszym są proste ścieżki i cykle niezawierające wierzchołków rozgałęziających. Drugi odpowiada bardziej rozbudowanym strukturom, które zawierają przynajmniej jedną tranzycję rozgałęziającą. Sytuacja jest odwrotna w przypadku p-niezmienników, dla których to miejsca rozgałęziające pełnią rolę elementów odpowiedzialnych za stopień skomplikowania takiej podsieci.

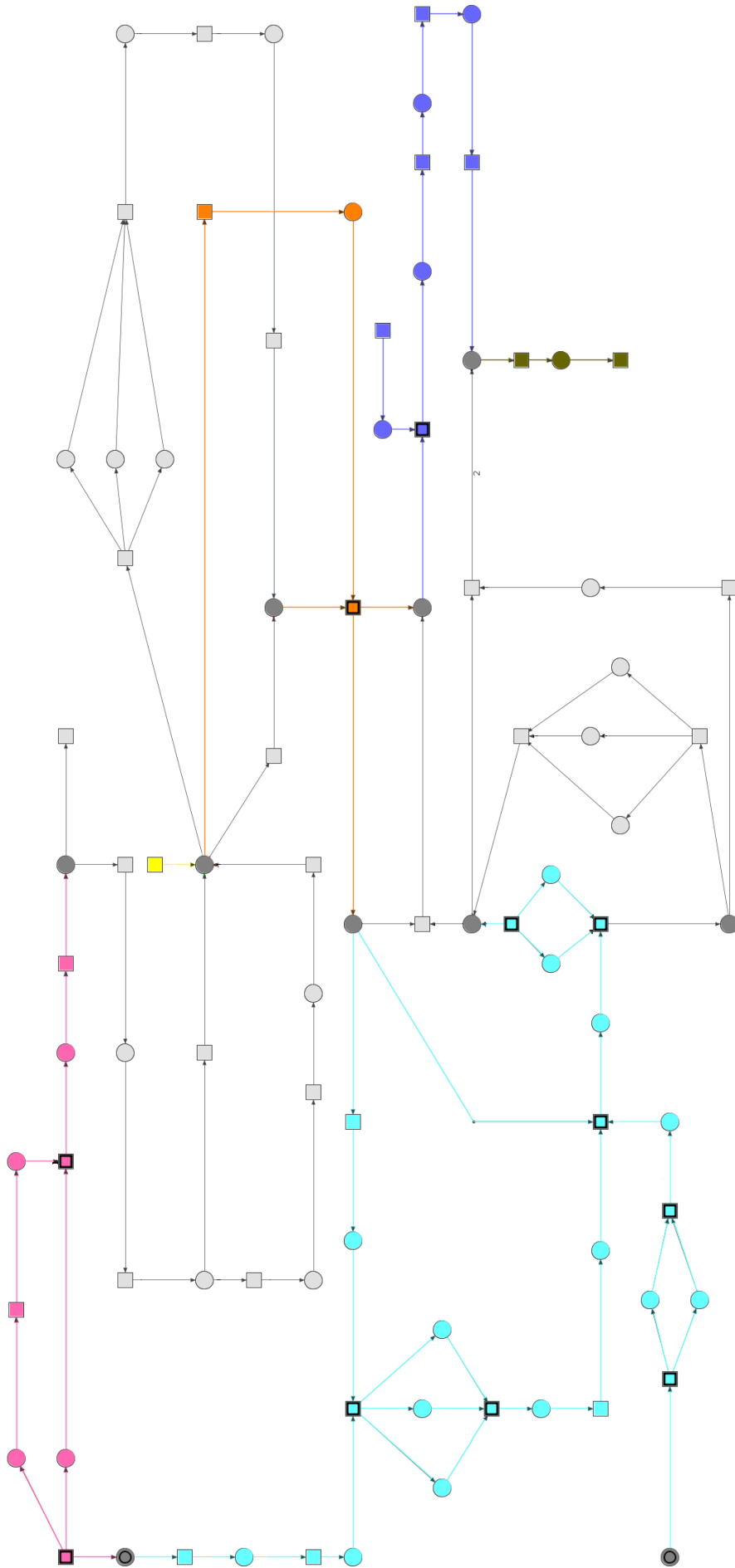
W algorytmie przedstawianym w pracy [67] do wyznaczania zbioru minimalnych niezmienników, bazującym na metodzie eliminacji Fouriera-Motzkina, można zaobserwować, że po przeprowadzeniu pierwszego kroku w algorytmie, uzyskujemy struktury indukowane przez parę R^{vbr} czyli tranzycje rozgałęziające wraz z ich końcówkami. To na ich bazie następnie wyznaczane są właściwe t-niezmienniki.

5.2.2 Wierzchołki rozgałęziające a podsieci ADT/MCT/t-sieci

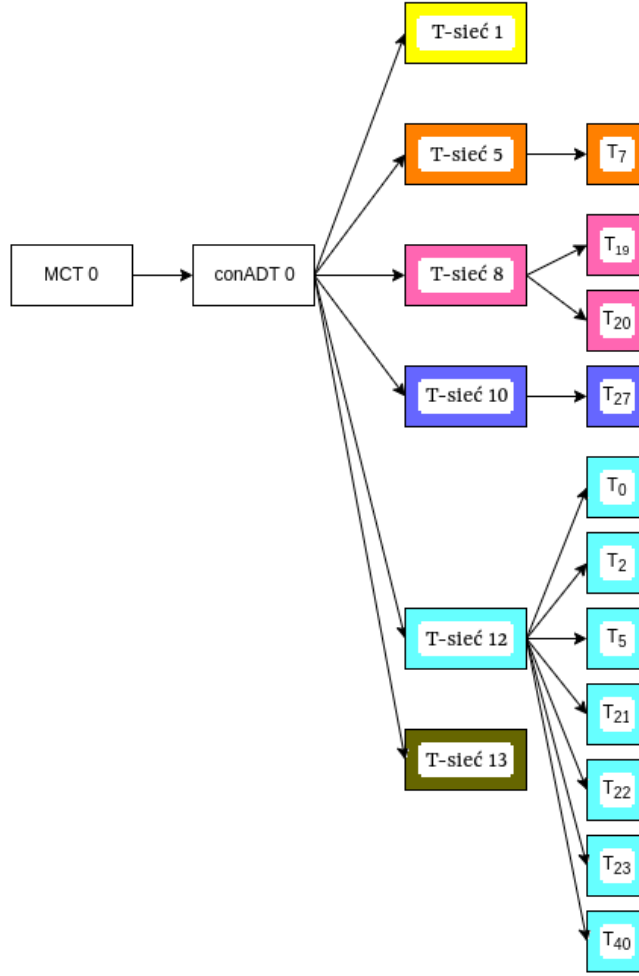
Podsieci indukowane przez zbiory maxADT (MCT) i conADT są wyznaczane na bazie zbioru minimalnych t-niezmienników, w wyniku czego współdzielą one charakterystyki związane z tranzycjami rozgałęziającymi. W zbiorach ADT są określone miejsca interfejsowe (ang. *interface places*), które odpowiadają obszarom styku pomiędzy różnymi zbiorami ADT. Każde miejsce interfejsowe jest miejscem rozgałęziającym, jednocześnie w strukturze indukowanej przez zbiór ADT, mogą znaleźć się miejsca rozgałęziające niebędące miejscami interfejsowymi (Rysunek 4.7 z Rozdziału 4).

Struktura oparta na t-sieci jest ograniczona miejscami, które są jednocześnie rozgałęziającymi i interfejsowymi $P^T = P_{IF}^T$. Nie zawiera ona innych miejsc rozgałęziających niebędących miejscami interfejsowymi. Pozostałe miejsca występujące w sieci nazywamy wewnętrznymi, a zbiór takich miejsc jest oznaczony jako P_{int}^T . Spełniają one poniższe warunki:

$$P_{int}^T = P^T \setminus P_{IF}^T \quad (5.13)$$



Rysunek 5.3: Na Rysunku przedstawiono przykładową sieć, wraz z jej największą podsiecią opartą o zbiór MCT 0, wraz z jej podziałem na fragmenty odpowiadające podsieciom t -sieci. Dodatkowo zaznaczono występujące w nich wierzchołki rozgałęziające: trzancyje za pomocą obramowania, a miejsca za pomocą ciemnoszarego koloru.



Rysunek 5.4: Powyżej przedstawiono za pomocą drzewa relację zawierania się tranzycji rozgałęziających w zdekomponowanej podsieci MCT 0.

$$\forall_{p \in P_{int}^T} |p^\bullet| \leq 1 \wedge |\bullet p| \leq 1 \quad (5.14)$$

w wyniku czego zawiera ona tylko trywialne miejsca przepływowe, wyjściowe i wejściowe.

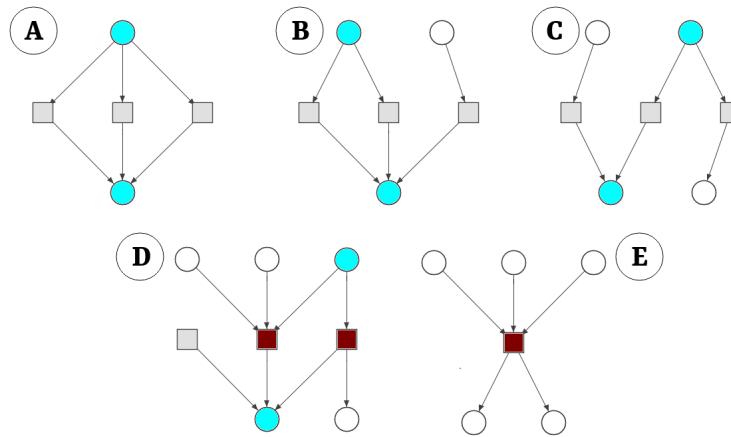
Wszystkie nietrywialne podsieci bazujące na zbiorach MCT, ADT czy t-sieci, zawierają przynajmniej jedną tranzycję rozgałęziającą. Jeśli tranzycja rozgałęziająca ma końcówkę będącą tranzycją, oba elementy będą należeć do tej samej struktury. Przypadkami takiej relacji są podsieci t-sieć 8 wraz z jej tranzycjami rozgałęziającymi t_{19} , t_{20} i t-sieć 12 wraz z tranzycjami rozgałęziającymi $t_0, t_2, t_5, t_{21}, t_{22}, t_{23}, t_{40}$ z Rysunków 5.3 i 5.4.

5.2.3 Wierzchołki rozgałęziające a podsieci funkcyjne

Mimo podobieństwa sieci funkcyjnych do opisywanych wcześniej typów dekompozycji, relacje i funkcje wierzchołków rozgałęziających są z ich perspektywy znacząco różne.

Aby przedstawić charakterystykę relacji pomiędzy wierzchołkami rozgałęziającymi wykorzystano przykład z Rysunku 5.5 zawierający przykładowe podsieci funkcyjne z zaznaczonymi tranzycjami i miejscami rozgałęziającymi. Pierwszą kluczową różnicą jest mniejsza istotność tranzycji rozgałęziających w porównaniu z perspektywą ADT/MCT/t-sieci. Nietrywialna podsieć nie musi zawierać już ani

jednej tranzycji rozgałęziającej t^{br} – przypadki A, B i C z Rysunku 5.5. Dodatkowo, podsieć zbudowana wokół tranzycji rozgałęziającej i nieposiadająca żadnego miejsca rozgałęziającego będzie przedstawiać prosty acykliczny graf – przypadek E z Rysunku 5.5. Nie może powstać podsieć funkcyjna zawierająca dwie tranzycje rozgałęziające i niezawierająca żadnego miejsca rozgałęziającego. Miejsca rozgałęziające nadal pełnią tą samą rolę co w podsięciach t-sieć i są wierzchołkami, na których dwie sąsiednie podsieci pokrywają się. Dodatkowo miejsce rozgałęziające może pełnić funkcję łącznika pomiędzy tranzycjami występującymi w ramach jednej podsieci – przypadki B, C, D z Rysunku 5.5.



Rysunek 5.5: Rysunek przedstawia kilka przykładowych sieci funkcyjnych wraz z zaznaczeniem miejsc rozgałęziających (kolor niebieski) i tranzycji rozgałęziających (kolor brązowy).

Rozdział 6

Grafilety w sieciach Petriego

6.1 Wprowadzenie

Rozdział ten poświęcony jest metodom porównywania sieci Petriego opartym o badanie rozkładu grafiletów w porównywanych modelach. Opisuje on zmiany jakie zostały wprowadzone w teorii dotyczącej grafiletów aby dostosować je do sieci Petriego. W rozdziale zawarto wyniki testów przeprowadzonych na metodach RGF i GDDA, do których wykorzystano generowane losowo sieci spełniające własności typowe dla modeli biologicznych. W przypadku metody GDDA przeprowadzono dodatkowo badanie jej zachowania w sieciach Petriego o niskiej gęstości, w celu potwierdzenia, czy problem niestabilności GDDA opisany w [107] zachodzi dla opracowanego w [120] wariantu grafiletów dedykowanych do sieci Petriego.

6.2 Modyfikacje struktur grafiletów

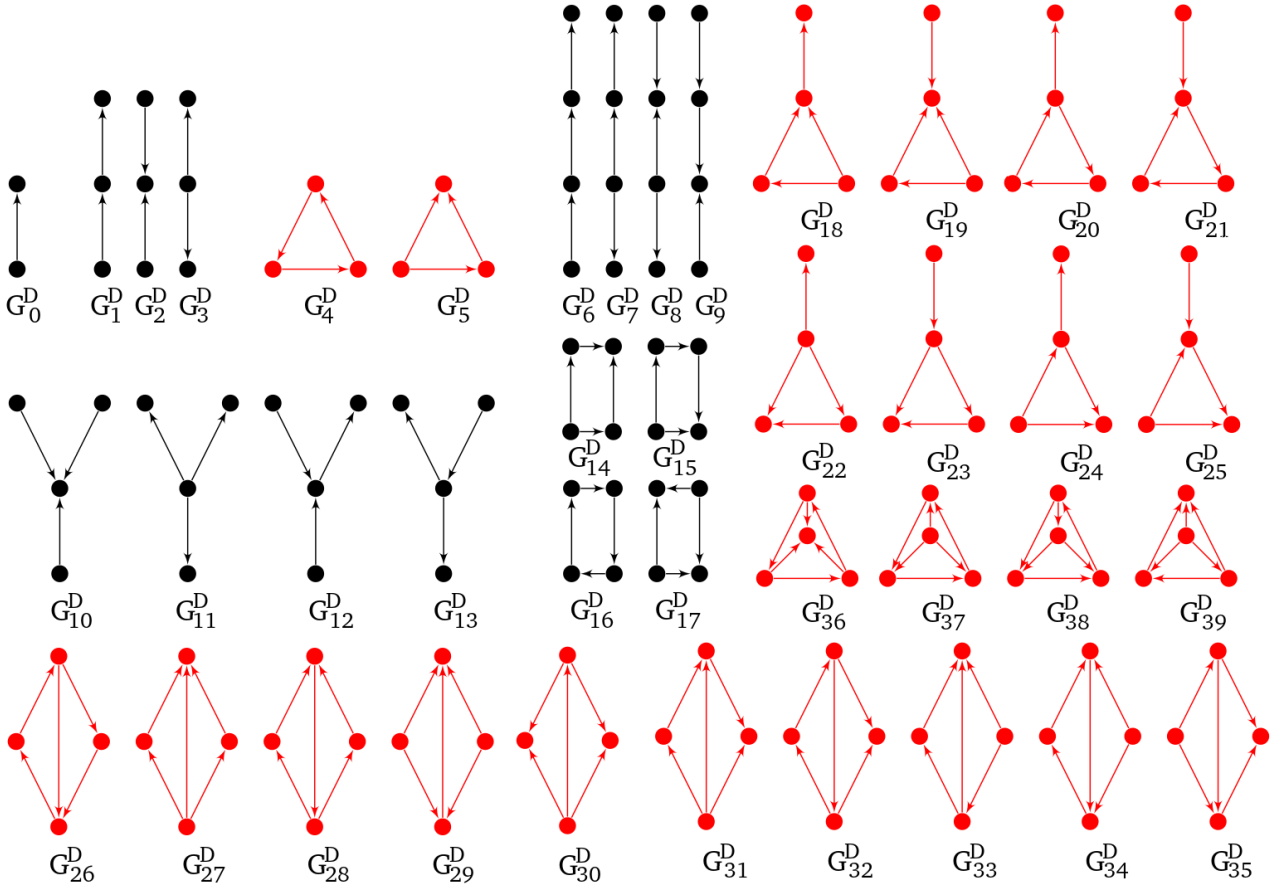
Aby wykorzystać w pełni możliwości, jakie daje użycie grafiletów do porównywania sieci Petriego, konieczne było dostosowanie ich do nowej postaci. Podobne działania były już w przeszłości wykonane np. dla digrafów [5, 110]. W przypadku klasycznych sieci Petriego konieczne było wprowadzenie dedykowanych grafiletów uwzględniających istnienie łuków, dwudzielność sieci, podział na dwa rodzaje wierzchołków i wagę łuków

Grafiley skierowane

Pierwszym etapem w adaptacji grafiletów do sieci Petriego było uwzględnienie skierowania łuków. Ten obszar był już wcześniej wnikliwie zbadany (więcej w podrozdziale 3.3.2) i oba ze wspomnianych w [5, 110] podejść możliwe były do wykorzystania przy budowie wariantu dla sieci Petriego. Ostatecznie wybrany został wariant skierowanych grafiletów bez uwzględniania dwukierunkowych łuków z następujących powodów.

Po pierwsze, ograniczenia jakie narzuca użycie łuków odczytu w modelach powodują, że nie one są kompatybilne z wieloma metodami analizy bazującymi na niezmiennikach. W efekcie można zaobserwować unikanie ich stosowania lub zastępowanie podsieciami, które symulują ich działanie.

Po drugie, użycie dwustronnych łuków niesie za sobą szereg wyzwań. Uwzględnienie pośród możliwych połączeń dwustronnego łuku powoduje znaczący wzrost liczby nowych grafiletów. W rezultacie



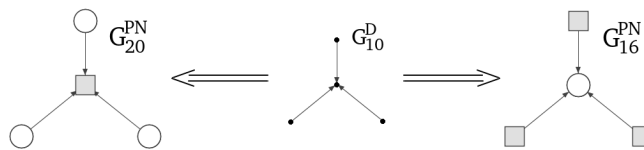
Rysunek 6.1: Zbiór wszystkich skierowanych grafletów o wielkości od 2 do 4 wierzchołków, nie zawierających dwukierunkowych łuków. Kolorem czarnym zaznaczono graflety będące sieciami dwudzielnymi.

konieczne jest operowanie na liczniejszych grafletach mniejszych rozmiarów. Użycie grafletów o dwustronnych łukach zwiększyłoby możliwości detekcyjne dla sieci zawierających łuki odczytu, kosztem sieci nieposiadających ich, które występują częściej w analizach prezentowanych w literaturze.

Rozszerzenie o łuki odczytu mimo ograniczeń jakimi są obciążone, zasługuje na uwagę w przyszłości.

Dwudzielność

Kolejną własnością sieci Petriego wymuszającą zmiany na strukturze grafletów jest dwudzielność. Część ze skierowanych grafletów zawiera w sobie nieskierowany cykl o nieparzystej długości jak np. graflety G_4^D , G_{20}^D z Rysunku 6.1, w wyniku czego nie wystąpią w sieci dwudzielnej. Tylko digrafony o numerach z zakresów 0–3 i 6–17 są możliwe w grafach dwudzielnych (co zaprezentowano kolorem czarnym na Rysunku 6.1).



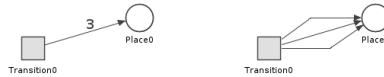
Rysunek 6.2: Przykład pokazujący rozróżnienie typów wierzchołków dla digrafletu i utworzenie dwóch pn-grafletów dedykowanych do sieci Petriego.

Sieci Petriego posiadają dwa typy wierzchołków i muszą być one rozróżniane za pomocą grafletów, aby poprawnie wykrywać podobieństwo. W efekcie na bazie każdego z digraflatów, jakie mogą wystąpić w dwudzielnym grafie, zostały utworzone dwa graflety dedykowane do sieci Petriego (pn-graflety) – przykład przedstawiono na Rysunku 6.2. Wyjątkiem jest digraflat G_{15}^D będący cyklem, który po zamianie typu wierzchołka na przeciwny pozostaje w automorfizmie z grafletem sprzed zmiany typów wierzchołków (jest to obserwowane dla wszystkich cyklicznych grafletów).

Multigrafy

Ostatnią charakterystyką, która wymaga zmian w grafletach, jest uwzględnienie wielokrotności połączeń pomiędzy wierzchołkami. Stworzenie nowych grafletów nawet dla k -ograniczonych równoległych łuków nie jest praktyczne ze względu na szybki przyrost liczby potencjalnych grafletów. Kompromisem może być wybranie pewnego podzbioru grafletów o równoległych łukach, które reprezentują pożądane struktury a następnie użycie ich przy analizie ich rozkładu w sieci.

Warto zaznaczyć, że graflety o pojedynczych łukach wciąż mogą zostać użyte do analizy multigrafów, jeśli przyjmujemy definicje podgrafów indukowanych krawędziowo, a nie jak w przypadku oryginalnej definicji grafletów, podgrafów indukowanych wierzchołkowo.



Rysunek 6.3: Dwie formy reprezentacji równoległych połączeń.

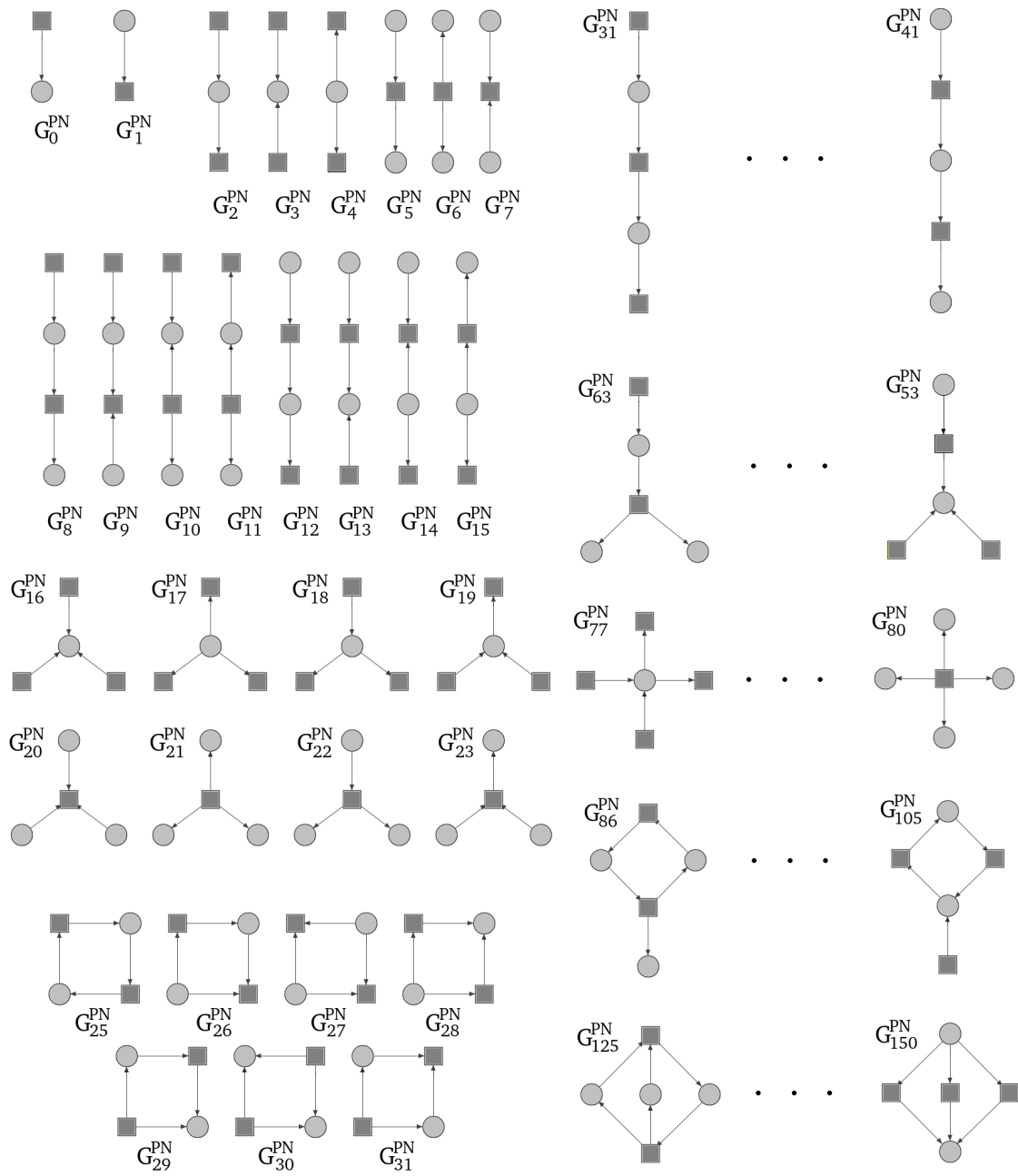
W efekcie istnieją trzy podejścia interpretacji równoległych łuków przy użyciu grafletów o pojedynczych łukach. Zostaną one przedstawione na przykładzie z Rysunku 6.3 i różnych odczytów liczby grafletów z ważonego łuku jaki się w nim znajduje:

- 1) graflet g_0^{pn} występuje w niej tylko jeden raz – waga jest pomijana, a łuk jest traktowany jako pojedynczy.
- 2) graflet g_0^{pn} występuje trzy razy – zliczane jest występowanie podgrafów indukowanych krawędziowo.
- 3) graflet g_0^{pn} nie występuje w grafie – jest to najbardziej konserwatywne podejście przy szukaniu podgrafów indukowanych wierzchołkowo, które nie znajduje żadnego wystąpienia.

Decyzja o wyborze jednego z zaprezentowanych podejść będzie zależna od kontekstu biologicznego porównywanych sieci, istotności wielokrotnych połączeń oraz ich liczby wystąpień.

pn-graflety

Na potrzebę odróżnienia od innych istniejących wariantów grafletów w artykule [120] wprowadzone zostało określenie *pn-grafletów* dla struktur dostosowanych do sieci Petriego. Otrzymano w ten sposób 151 grafletów o rozmiarze od 2 do 5 wierzchołków. Dla każdego z opracowanych pn-grafletów wyznaczono orbity, co pozwoliło na wykorzystanie metod porównywania które są o nie oparte, jak np. GDDA. W 151 pn-grafletach znajdują się 592 orbity. Dokładne informacje na temat liczby pn-grafletów i ich orbit znajdują się w Tabeli 6.1.



Rysunek 6.4: Przedstawienie pn-grafletów. Dla przypadków o rozmiarze 5 ($G_{31}^{PN} - G_{150}^{PN}$) ze względu na ich dużą liczbę, przedstawiono pojedyncze przykłady dla każdej podrodziny.

Tabela 6.1: Przyrost liczby pn-grafletów i ich orbit wraz ze wzrostem wielkości pn-grafletów

Rozmiar grafletu	Graflety		Orbity	
	Liczba	Suma (Σ)	Liczba	Suma (Σ)
2-wierzchołki	2	2	4	4
3-wierzchołki	6	8	14	18
4-wierzchołki	23	31	72	90
5-wierzchołki	120	151	502	592

Wszystkie pn-graflety o rozmiarze do 4 i przykłady dla pn-grafletów o rozmiarze 5 przedstawione są na Rysunku 6.4. Kwestią otwartą pozostaje wykorzystanie większych pn-grafletów. Tempo przyrostu pn-grafletów jest mniejsze niż dla digrafletów a większe niż dla grafletów nieskierowanych. Strukturalnie pn-graflety reprezentują 10 podzbiorów odpowiadających nieskierowanym grafletom występujących w sieciach Petriego.

W następnym podrozdziale zostaną przedstawione algorytmy i metryki porównywania sieci bazujące na pn-grafletach wraz z opisem zmian w ich zachowaniu względem nieskierowanych grafletów.

6.3 Wyznaczanie rozkładu

Przed rozpoczęciem wszelkich operacji mających na celu porównanie dwóch sieci konieczne jest określenie rozkładu pn-grafletów w każdej z nich. Jest to proces obciążony największym kosztem obliczeniowym, który jednocześnie jest zależny od rozmiaru użytych pn-grafletów i stopni wierzchołków w sieci. Metody porównywania bazujące na orbitach wymagają użycia osobnych algorytmów do ich wyznaczania. Wyznaczenie rozkładu wszystkich orbit reprezentuje bardziej złożony problem w porównaniu ze znalezieniem samych grafletów. Mimo że popularne jest użycie heurystyk do wyznaczenia grafletów i ich orbit, ze względu na stosunkowo małe rozmiary typowych sieci Petriego (od kilkudziesięciu do kilkuset wierzchołków), zastosowanie algorytmów dokładnych jest możliwe w praktycznych zastosowaniach. Oczywiście wraz z użyciem pn-grafletów o rozmiarze większym niż 5, ta sytuacja może ulec zmianie i wymóc zmianę podejścia na preferującą heurystykę.

W trakcie badań opracowane zostały dwa algorytmy do zliczania pn-grafletów. Pierwszy z nich powstał jako rozszerzenie algorytmu zliczającego orbitę na rzecz metryki GDDA opisanej w podrozdziale 6.5.1. Następuje to przez modyfikację znalezionych mapowań orbit do unikalnych grafletów. Ponieważ metoda powstała jako uzupełnienie prac nad GDDA, jest zależna od wyznaczenia rozkładu orbit, co ogranicza ją do przypadków wykonania pełnych analiz z użyciem różnych metod opartych o graflety.

Druga z opisywanych metod powstała później jako dedykowane narzędzie do wyznaczania rozkładu grafletów. Bazuje ona na zbiorze łuków występujących w badanej sieci. Każdy łuk odpowiada jednemu z dwóch grafletów G_0^{PN} lub G_1^{PN} określającymi kierunek łuku pomiędzy tranzycją a miejscem. Następnie w wyniku przeszukania w głąb odnajdywane są graflety o danym rozmiarze. Algorytm jest powtarzany aż do znalezienia wszystkich grafletów o poszukiwanych rozmiarach (Algorytm 1).

6.4 RGF

Zastosowanie RGF dla sieci Petriego nie wymagało żadnych zmian w sposobie działania metryki opracowanej dla nieskierowanych grafletów. Zmianie uległ jedynie zbiór wejściowy operujący teraz na 151 pn-grafletach. Tym samym zostały odziedziczone wszystkie zalety i wady, obserwowane przy

Algorytm 1 Algorytm poszukiwania grafletów

```

1: procedure FINDGRAPHLETS(graph, n)
2:   for a ∈ graph.arcs do
3:     result += searchGraphlets(graph, a, n,  $\emptyset$ ,  $\emptyset$ )
4:   end for
5: end procedure
6: procedure SEARCHGRAPHLETS(graph, arc, size, subnet, visited, graphlets)
7:   visited.add(arc)
8:   subnet.add(arc)
9:   if length(subnet) == size then
10:    graphlets.add(copySubgraph(subnet))
11:   else
12:     for neighbor in graph.getNeighbors(arc) do
13:       if !visited.contains(node) then
14:         searchGraphlets(graph, neighbor, size, subnet, visited, graphlets)
15:       end if
16:     end for
17:     subnet.removeLast()
18:     visited.remove(arc)
19:   end if
20:   return subgraphs
21: end procedure

```

użytkowaniu RGF dla sieci interakcji międzyproteinowych (ang. Protein-Protein Interaction network – PPI).

Algorytm 2 Procedura wyznaczania DRGF

```

1: procedure CALCRGF(G)
2:   gG = findsGraphlets(G)
3:   gH = findsGraphlets(H)
4:   for i = 0; i < 151 do
5:     nG[i] = getNumberOf(gG, i)
6:     nH[i] = getNumberOf(gH, i)
7:     tG = sum(gG)
8:     tH = sum(gH)
9:     rgf += Math.abs( $\frac{g_G(i)}{t^G} - \frac{g_H(i)}{t^H}$ )
10:   end for
11: end procedure

```

Po wyznaczeniu wszystkich grafletów występujących w dwóch porównywanych sieciach, uzyskanie wartości *RGF* sprowadza się do przeiterowania po wszystkich typach grafletów. Jednocześnie wyznaczone są wektory n^G i n^H odpowiadające rozkładowi grafletów dla sieci *G* i *H*. Wektory te używane są przy wizualizacji wyników w postaci diagramu częstotliwości, będącego uzupełnieniem dla znormalizowanej wartości *RGF* (Algorytm 2). Porównanie rozkładów pozwala na oszacowanie skali i formy różnic zachodzących pomiędzy sieciami *G* i *H*.

Przeprowadzona została analiza porównawcza pomiędzy rozkładem pn-grafletów, a największym wspólnym podgrafem (NWP). Podczas wyznaczania podobieństwa podział dwóch sieci na część wspólną i podsieci występujące jedynie w jednej z nich jest intuicyjny, choć niekoniecznie łatwy do uzyskania (problem znalezienia maksymalnego wspólnego podgrafu). Relacja pomiędzy rozkładem pn-grafletów w NWP a dwoma porównywanymi sieciami zostanie przedstawiona na następującym przykładzie.

Niech N będzie wektorem wystąpień pn-grafletów w nadsieci powstałej z sieci Petriego G i H .
Zatem:

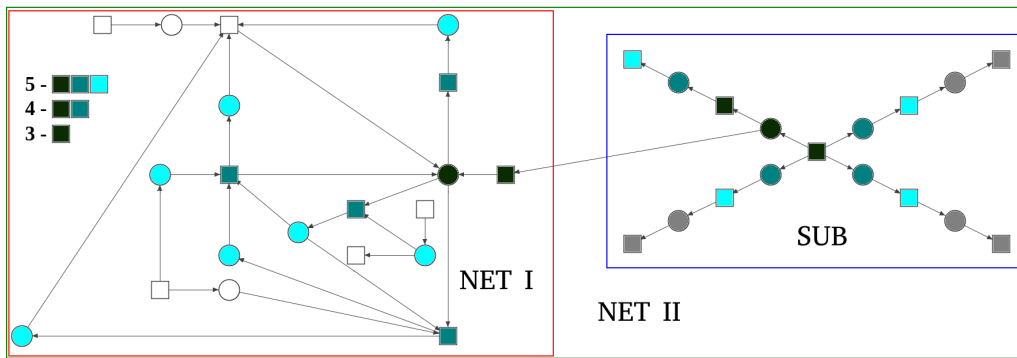
$$N = N^{NWP} + (N^{diff} + N^{con}), \quad (6.1)$$

gdzie:

- N^{NWP} – wektor zawierający graflety występujące w NWP,
- N^{Diff} – wektor zawierający graflety z podgrafu $Diff$, powstałego z wierzchołków występujących tylko w jednej z sieci,
- N^{con} – wektor zawierający graflety powstałe w wyniku połączenia łukami komponentów $Diff$ i NWP .

Należy zwrócić uwagę, że suma wektorów N^{Diff} i N^{NWP} nie odpowiada wektorowi N , a łącznej sieci składającej się z komponentów NWP i $Diff$. Każdy łuk występujący w bazowej sieci a niewystępujący w NWP i $Diff$, odpowiada za utworzenie zbioru nowych pn-grafletów, których liczba i rodzaj zależy od miejsca połączenia komponentów.

Na Rysunku 6.5 przedstawiono trzy sieci: Net-I, Net-II i Sub. Net-I zawiera się w całości w sieci Net-II, natomiast Sub przedstawia strukturę występującą jedynie w Net-II. Rozkład pn-grafletów został przedstawiony w Tabeli 6.2. Dzięki wspomnianej wcześniej relacji zawierania, Net-I odpowiada NWP w obu sieciach. Po wyznaczeniu rozkładu osobno dla Net-II i Sub możliwe jest określenie zbioru pn-grafletów, które powstały w wyniku połączenia tych komponentów. Na Rysunku 6.5 kolorami zaznaczono zasięg nowo powstałych pn-grafletów (kolejno dla rozmiaru 3, 4 i 5). Obrazuje to zasięg detekcji struktur, jaki daje użycie pn-grafletów konkretnych rozmiarów. Gdyby łuk łączący komponenty występował pomiędzy innymi wierzchołkami, indukowałby on nowy zbiór pn-grafletów. W przypadku, gdy podsieci odpowiadające obszarowi pokrytemu przez nowe pn-graflety są izomorficzne, ich wektory grafletów N^{con} będą identyczne.



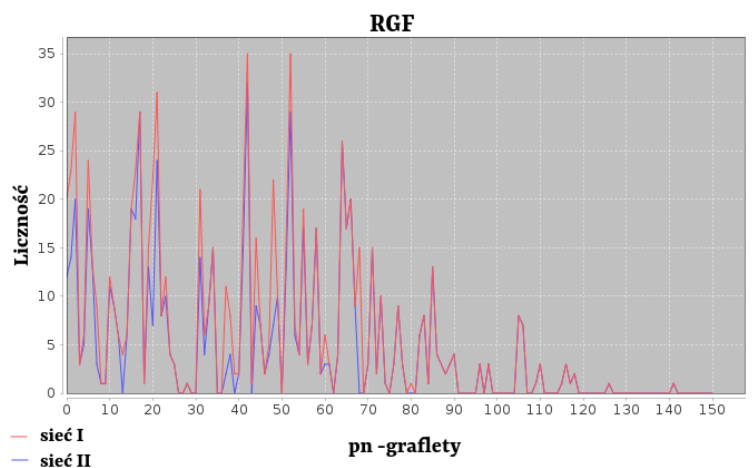
Rysunek 6.5: Sieć Net-II wraz z zawierającą się w niej siecią Net-I oraz obszarem na którym pojawiły się nowe graflety w wyniku powstania połączenia Net-I z Sub. Zasięg występowania nowych grafletów zaznaczono kolorami z rozróżnieniem na graflety rozmiarów 3-, 4-, 5-

Na diagramie z Rysunku 6.6 przedstawiono relację pomiędzy rozkładami pn-grafletów. Dla większości pn-grafletów rozmiaru 5 widać idealne pokrycie ich rozkładu. Dla pozostałych widać, że liczby wystąpień pn-grafletów z Net-I zawierają się w Net-II. Taki przebieg jest przesłanką do zawierania się struktur, ale nie musi go potwierdzić.

Tabela 6.2: Rozkład grafletów z podziałem na porównywane sieci i różnice pomiędzy nimi.

Grafiety	Net-I	Net-II	Sub	Dystans	Nowe
0	20	12	8	8	0
1	23	14	8	9	1
2	29	20	8	9	1
3	3	3	0	0	0
4	6	5	0	1	1
5	24	19	4	5	1
6	13	13	0	0	0
7	9	3	6	6	0
8	1	1	0	0	0
9	1	1	0	0	0
10	12	11	0	1	1
11	9	9	0	0	0
12	6	6	0	0	0
13	4	0	4	4	0
14	6	6	0	0	0
15	19	19	0	0	0
16	23	18	4	5	1
17	29	29	0	0	0
18	1	1	0	0	0
19	15	13	0	2	2
20	22	7	12	15	3
21	31	24	4	7	3
22	8	8	0	0	0
23	12	10	0	2	2
24	4	4	0	0	0
25	3	3	0	0	0
28	1	1	0	0	0
31	21	14	4	7	3
32	6	4	0	2	2
33	9	9	0	0	0
34	15	15	0	0	0
37	11	2	6	9	3
38	8	4	0	4	4
39	2	0	0	2	2
40	2	2	0	0	0
41	19	15	0	4	4
42	35	32	0	3	3
43	1	0	0	1	1
44	16	9	0	7	7
45	7	7	0	0	0
46	2	2	0	0	0
47	5	4	0	1	1
48	22	7	12	15	3
49	10	10	0	0	0
51	16	13	0	3	3
52	35	29	0	6	6
53	7	6	0	1	1
54	4	4	0	0	0
55	19	17	0	2	2
56	3	3	0	0	0
57	7	7	0	0	0
58	17	17	0	0	0
59	2	2	0	0	0
60	6	3	0	3	3
61	3	3	0	0	0
63	4	4	0	0	0
64	26	26	0	0	0
65	17	17	0	0	0
66	20	20	0	0	0

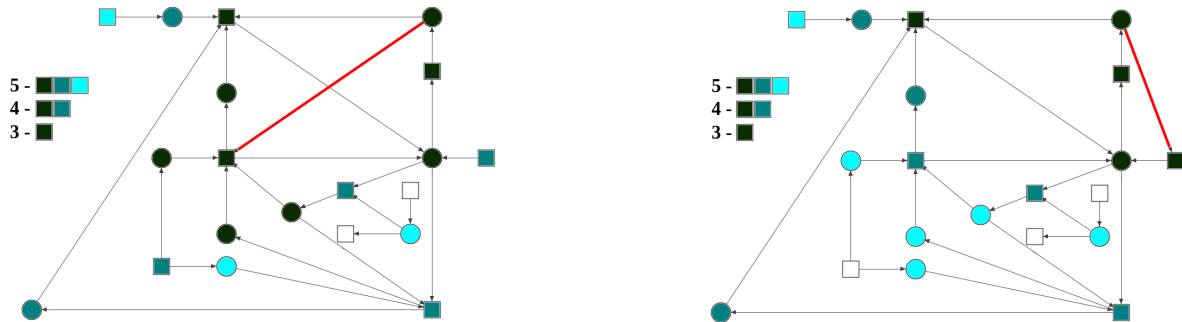
Grafiety	Net-I	Net-II	Sub	Dystans	Nowe
67	9	9	0	0	0
68	15	0	12	15	3
70	3	3	0	0	0
71	15	15	0	0	0
72	2	2	0	0	0
73	10	10	0	0	0
74	1	1	0	0	0
76	3	3	0	0	0
77	9	9	0	0	0
78	3	3	0	0	0
80	1	0	1	1	0
82	6	6	0	0	0
83	8	8	0	0	0
84	1	1	0	0	0
85	13	13	0	0	0
86	4	4	0	0	0
87	3	3	0	0	0
88	2	2	0	0	0
89	3	3	0	0	0
90	4	4	0	0	0
96	3	3	0	0	0
98	3	3	0	0	0
105	8	8	0	0	0
106	7	7	0	0	0
109	1	1	0	0	0
110	3	3	0	0	0
115	1	1	0	0	0
116	3	3	0	0	0
117	1	1	0	0	0
118	2	2	0	0	0
126	1	1	0	0	0
141	1	1	0	0	0



Rysunek 6.6: Rozkład grafletów dla sieci Net-I i Net-II.

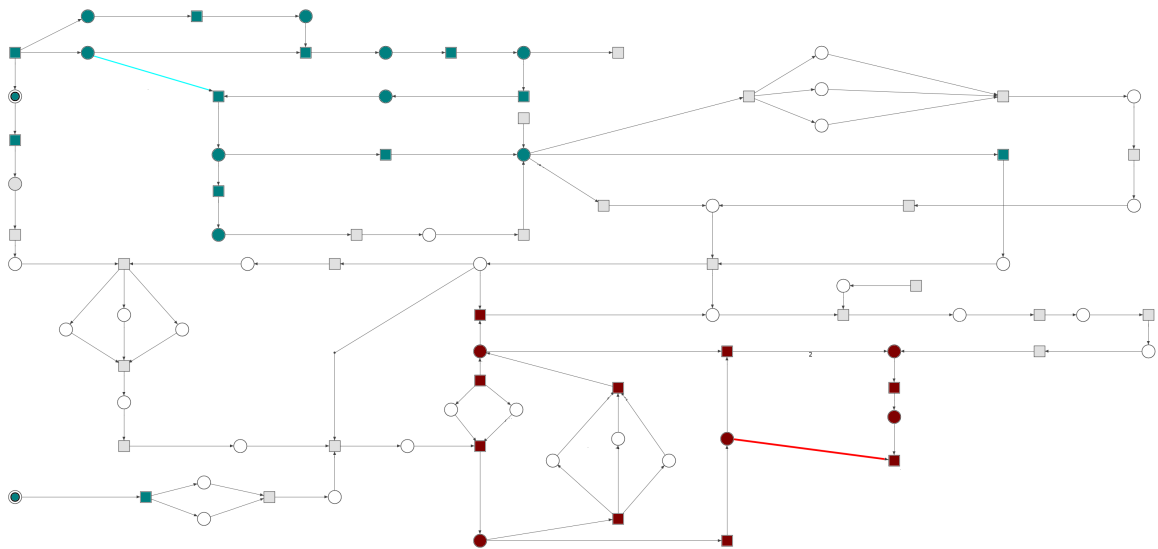
Na Rysunku 6.7 przedstawiono dwie sieci Net-III i Net-IV. Od sieci Net-I z poprzedniego przykładu różnią się one dodaniem jednego dodatkowego łuku (zaznaczonego na czerwono). Obrazują one więc przypadki, w których dystans grafowy (łukowy) między sieciami Net-I i Net-III, oraz Net-I i Net-IV wynosi 1, natomiast RGF wynosi kolejno 55, 51 i 41, 53. Różnica w wynikach uzyskanych za pomocą danych metod jest większa niż można by się spodziewać. Wynika to z nowo powstałych pn-grafletów wykorzystujących dodatkowy łuk. Pokrywają one sobą niemal cały obszar sieci z pominięciem kilku tranzycji.

W wyniku użycia RGF uzyskujemy pewną informację o lokalizacji różnic występujących pomiędzy porównywanymi sieciami. Zaznaczyć należy, że wystąpienie różnicy w obszarze sieci o wysokiej grafowej gęstości będzie objawiał się większą liczbą nowych grafletów i ich rozmiarem. To następnie pozwala na próbę interpretacji wpływu różnic strukturalnych na przepływy występujące wewnątrz sieci. W rezultacie wynik zwracany przez RGF jest bliższy w interpretacji do kosztu transformacji grafów zwracanego przez GED niż do dystansu grafowego.



Rysunek 6.7: Przykład zmian w rozkładzie pn-grafletów po dodaniu pojedynczego łuku do losowej sieci.

Poprzedni przykład pokazuje wpływ pojedynczego łuku różniącego relatywnie małe sieci. Obszar, na którym pojawiły się nowe pn-graflety (zaznaczony na Rysunku 6.7 za pomocą kolorów), jest duży w stosunku do rozmiaru całych sieci. Na Rysunku 6.8 przedstawiono podobne zmiany, ale na przykładzie znacznie większego modelu układu biologicznego.



Rysunek 6.8: Zasięg nowopowstałych grafletów rozmiaru 5 w wyniku dodania do sieci Petriego dwóch pojedynczych łuków.

6.5 GDDA

Prace nad dostosowaniem grafletów do sieci Petriego rozpoczęte zostały od implementacji i testów metody GDDA, która została wstępnie oceniona jako bardzo obiecująca. W wyniku tego najpierw

powstał algorytm odnajdywania wszystkich orbit, co jest odwrotnym podejściem wobec tych stosowanych dla nieskierowanych i skierowanych sieci. Ponieważ początkowym celem prac było sprawdzenie, czy GDDA i pn-graflety nadają się do porównywania systemów biologicznych pod postacią sieci Petriego, kwestia wysokiej wydajności algorytmu nie była priorytetem. W efekcie powstał wykładniczy algorytm dokładny, który mimo to okazał się wystarczający dla zdecydowanej większości testowanych sieci (podobnie jak część algorytmów opisanych w literaturze [55], jego czasowa złożoność jest silnie zależna od stopni wierzchołków w porównywanych sieciach).

6.5.1 Wyznaczanie GDDA

Do wyznaczenia GDDA konieczne jest wykonanie następujących kroków:

1. Wyznaczenie dla każdego z wierzchołków liczby dotykających go orbit,
2. Wyznaczenie macierzy F_r zawierającej k-pojawień orbit,
3. Wyznaczenie macierzy GDD będącej różnicą macierzy F_{r_G} i F_{r_H} ,
4. Standaryzacja i doprowadzenie do postaci GDDA.

Krok I (Zliczanie orbit) Dla każdego z wierzchołków w sieci sprawdzane jest czy i ile razy orbita o_i pojawia się w nim (Algorytmy 3 i 4). Jako że w pn-grafletach orbity są ściśle powiązane z konkretnym typem wierzchołków, sprawdzane są tylko te współdzielące typ (dla pn-grafletów o rozmiarze 5 istnieje 296 orbit powiązanych z tranzycjami i 296 orbit powiązanych z miejscami).

Algorytm 3 Wyznaczanie rozkładu orbit

Require: $G = (V, E)$

Ensure: Znajdź wszystkie wystąpienia orbit w G

```

1: for  $v_j \in V$  do
2:   for  $o_i \in O$  do
3:      $K[v_j][o_i] = \text{findOrbits}(v_j, o_i);$ 
4:   end for
5: end for
```

Algorytm 4 Metoda findOrbits()

Require: o_i, v_j

Ensure: Znajdź ile razy o_i dotyka v_j

```

1:  $\text{listUnfinSubnets.add}(\text{newSubnet}(v_j))$ 
2: while  $\text{listUnfinSubnets.size}() > 0$  do
3:    $\text{subUn} = \text{listUnfinSubnets.get}(0)$ 
4:   if condition then
5:      $X.\text{add}(\text{subUn});$ 
6:   else
7:      $\text{listUnfinSubnets} += \text{getNeighb}(\text{subUn})$ 
8:   end if
9: end while
10:  $X = \text{checkMatching}(X)$ 
11:  $X = \text{getUnique}(X)$ 
```

Rezultatem jest macierz K o rozmiarach $[n \times o]$ zawierająca informacje o tym, ile orbit konkretnego typu dotyka dany wierzchołek.

Operacja odnalezienia, ile razy orbita o_i występuje w wierzchołku v_j jest rdzeniem całego procesu. Uzyskiwana jest poprzez przeszukanie sieci włąb, rozpoczynane z v_j . Uzyskiwana struktura jest porównywana z wygenerowanym wcześniej pn-grafletem, do którego należy orbita o_i .

Wszystkie znalezione struktury są dodatkowo sprawdzane pod względem poprawności mapowania z grafletem ($checkMatching(X)$), a następnie redukowane są duplikaty ($getUnique(X)$).

Krok II Na podstawie macierzy K konieczne jest zliczenie ile razy każda z orbit o_i występuje we wszystkich wierzchołkach dokładnie $1, 2, \dots, k$ razy. Do tego wystarczy $|n| \cdot |o| \cdot |k|$ krotne przeiterowanie przez macierz K . W celu uzyskania macierzy GDD konieczne jest wyznaczenie różnicy macierzy F_G i F_H , a następnie sprowadzenie poszczególnych pozycji w macierzy do wartości bezwzględnej (Algorytm 5).

Algorytm 5 Wyznaczanie GDD

```

1: procedure VECTORORBIT( $G$ )
2:   for int orb = 0; orb < d.length; orb++ do
3:     double[] ss = new double[d[orb].length];
4:     for int k = 1; k < d[orb].length; k++ do
5:       double dd = (double) d[orb][k];
6:       ss[k] = dd / (double) k;
7:       s[orb] = ss;
8:     end for
9:   end for
10:  for int orb = 0; orb < d.length; orb++ do
11:    t[orb] = DoubleStream.of(s[orb]).sum();
12:  end for
13:  for int orb = 0; orb < n.length; orb++ do
14:    for int k = 0; k < n[orb].length; k++ do
15:      n[orb][k] = s[orb][k] / t[orb];
16:    end for
17:  end for
18: end procedure

```

Krok III Ostatnim krokiem jest przeprowadzenie operacji normalizacji na macierzy GDD, aby sprowadzić ją do wartości GDDA (Algorytm 6).

6.5.2 Zachowanie GDDA w sieciach Petriego

Kwestia średniego podobieństwa – GDDA

W zależności od wybranego podejścia porównywania sieci, istnieją różne sposoby przedstawienia jego skali. W większości przypadków są one ściśle powiązane z grupą algorytmów porównywania (Tabela 3.2). Część z nich przedstawia rezultat porównania w łatwy do interpretacji sposób np. dystans grafowy (liczba różnic) czy największy wspólny podgraf (procentowy poziom podobieństwa struktur). W przypadku metod opartych o graflety i ich orbity (np. GDD i GDDA) nietrywialnym jest określenie czy porównywane modele są podobne oraz gdzie leży granica, od której można jednoznacznie potwierdzić

Algorytm 6 Wyznaczanie GDDA

```

1: procedure CALCGDDA( $d$ )  $\triangleright d$  is a GDDA matrix
2:   for int orb = 0 ; orb < orbNumber ; orb++ do
3:     double  $d_i$  = 0;
4:     for int k = 0 ; k < maxk ; k++ do
5:       if k >= nG[orb].length then
6:          $d_i$  += Math.pow(0-nH[orb][k],2);
7:       else if k >= nH[orb].length then
8:          $d_i$  += Math.pow(nG[orb][k]-0,2);
9:       else
10:         $d_i$  += Math.pow(nG[orb][k]-nH[orb][k],2);
11:      end if
12:    end for
13:  end for
14: end procedure

```

podobieństwo. Aby móc odpowiedzieć na to pytanie, konieczne jest sprawdzenie, jaką wartość przyjmuje GDDA dla dwóch przypadkowych (losowych) sieci. Wynik pozwoli określić punkt odniesienia dla interpretacji uzyskiwanych rezultatów.

Aby móc odpowiedzieć na postawione powyżej pytanie, najpierw należy ustalić, jak wygląda statystycznie typowy model wyrażony za pomocą sieci Petriego spotykany w zastosowaniach biologicznych, ale nie tylko. Obserwując modele pojawiające się w publikacjach można określić szereg cech powszechnie współdzielonych pomiędzy nimi. Parametry testowych sieci są następujące:

- 20-100 wierzchołków,
 - 10-50 miejsc,
 - 10-50 tranzycji,
- wielokrotne łuki dozwolone,
- brak miejsc wejściowych i wyjściowych,
- przynajmniej po jednej tranzycji wejściowej i wyjściowej lub brak obu.

W porównaniu do modeli PPI sieci Petriego nie osiągają tak dużych rozmiarów. W przypadku interakcji między białkami często brane są pod uwagę setki lub tysiące białek co przekłada się na liczbę wierzchołków w sieciach PPI. Natomiast sieci Petriego są używane do modelowania innych typów zjawisk co ma wpływ na ich mniejszy rozmiar.

Dodatkowo stosunek pomiędzy liczbą wystąpień różnych typów wierzchołków może mieć wpływ na wzrost występowania poszczególnych grafletów. W modelach znalezionych w literaturze zaobserwowana została stała choć niewielka nadreprezentacja tranzycji względem miejsc. Było to przesłanką do zbadania zachowania grafletów w sieciach gdzie balans pomiędzy typami wierzchołków jest zaburzony. Z tego powodu pośród wygenerowanych przypadków testowych pojawiły się modele posiadające aż pięciokrotną przewagę jednego typu wierzchołka nad drugim.

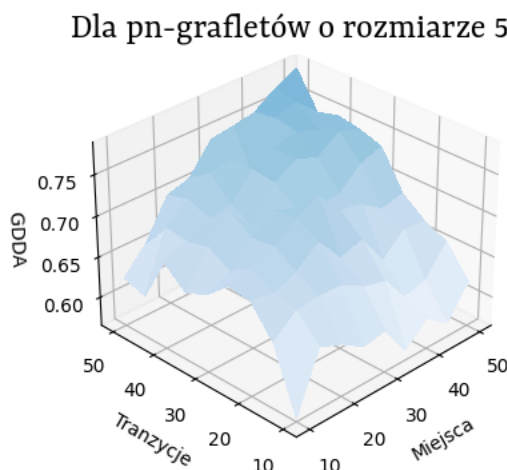
Część modeli przedstawianych za pomocą sieci Petriego charakteryzuje się koniecznością spełnienia konkretnych własności takich jak zwyczajność (ang. *ordinary* – gdzie wszystkie łuki mają wagę 1) i homogeniczność (ang. *homogenic* – wszystkie łuki wychodzące z miejsca mają tą samą wagę). Wpływa

to na występowanie ważonych łuków w sieciach i jest powszechnie spotykane w przemysłowych modelach maszyn stanowych wyrażonych za pomocą sieci Petriego. W przypadku zastosowań biologicznych takie ograniczenia nie są powszechnie stosowane (przykłady modeli spełniających powyższe własności można znaleźć w [44, 35, 51] a niespełniających w [35, 51, ?]), stąd pośród generowanych modeli mogą znaleźć się przypadki z jaki i bez wielokrotnych łuków.

Model wyrażony za pomocą sieci Petriego opisuje jedynie fragment większego systemu biologicznego. To, czy dany podproces, reakcja, czynnik w istotny sposób wpływa na modelowany system i powinien pojawić się w modelu jako wierzchołek jest subiektywną decyzją autora bazującego na wiedzy eksperckiej. Granicę modelu, czyli obszary w których następuje wymiana sygnałów (przyływu lub odpływu tokenów) pomiędzy modelem a jego niezamodelowanym otoczeniem, reprezentują tranzycje wejściowe i wyjściowe.

Granicę modelu mogłyby w szczególnych przypadkach reprezentować miejsca wejściowe i wyjściowe, z pewną znaczącą różnicą. Model zawierający miejsca wejściowe jest ograniczony przez skończoną liczbę tokenów (zasobów) znajdujących się w tych miejscach, w przeciwieństwie do tranzycji wejściowej, która ze stałym prawdopodobieństwem dostarcza tokeny do modelu. Jest to jednak podejście niespotykane pośród zastosowań biologicznych. Powodem może być problem (koszt) dokładnego określenia liczby zasobów, które przekładałyby się na liczbę tokenów.

Rezultat testu porównującego wygenerowane losowo sieci spełniających określone warunki został przedstawiony na Rysunku 6.9. Wynioskować z niego można, że średnie podobieństwo przedstawione w postaci GDDA wzrasta wraz z rozmiarem porównywanych modeli. Ma to bezpośrednie podstawy w tym, że w większych modelach statystycznie istnieje większa szansa na zaistnienie struktury odpowiadającej pewnemu grafletowi.



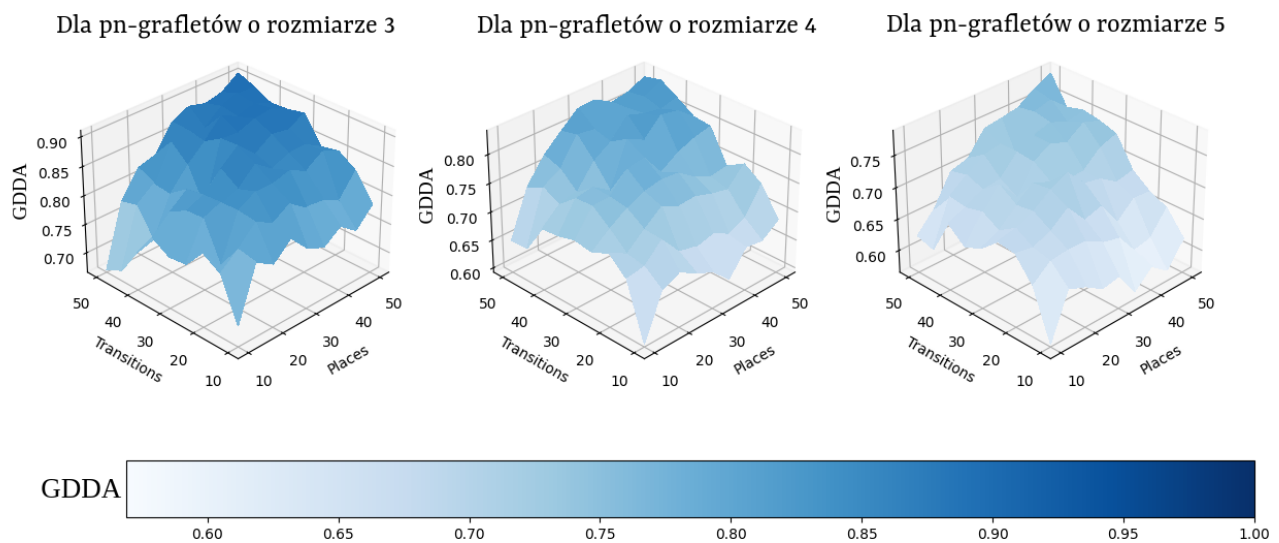
Rysunek 6.9: Podobieństwo losowych sieci spełniających wymagania modeli systemów biologicznych.

Efektom tej sytuacji jest konieczność innej interpretacji w zależności od rozmiaru sieci. Przykładowe dwie 20-wierzchołkowe sieci, dla których GDDA przyjmuje wartość 0,80 wykazują znacznie większe podobieństwo niż dwie 100-wierzchołkowe sieci o tej samej wartości GDDA.

Liczba grafletów a dokładność GDDA

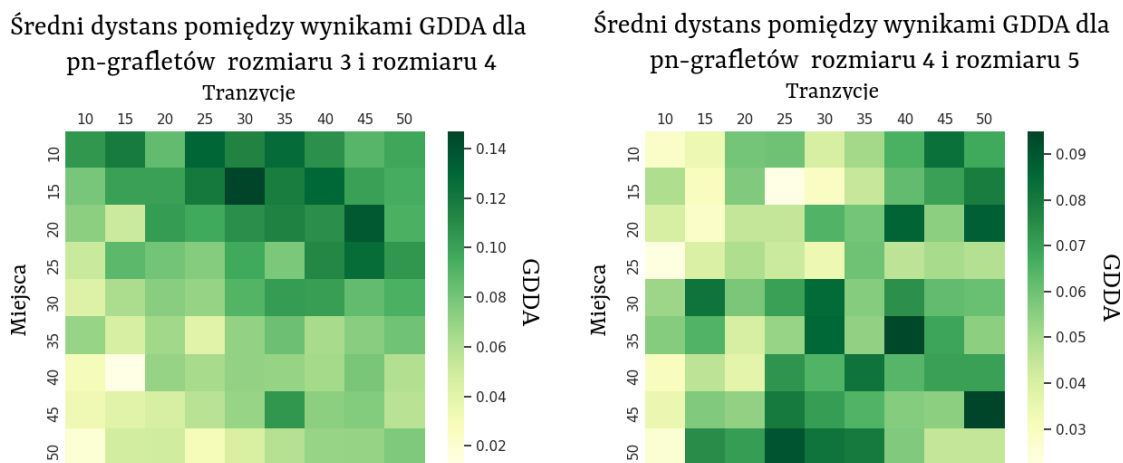
Radzenie sobie z nagłym wzrostem liczby grafletów wraz ze wzrostem ich rozmiaru jest integralną częścią metod analitycznych nieparujących na grafletach. Z tego powodu zaproponowano ograniczenie do grafletów opartych o 5 wierzchołków (4 w przypadku grafletów skierowanych). Reprezentują one kompromis pomiędzy liczbą zliczanych struktur, a możliwościami detekcji różnic, jakie daje użycie większych grafletów. Dlatego to właśnie graflety tych rozmiarów obserwujemy w praktycznych zastosowaniach, podczas gdy większe i przez to liczniejsze warianty obecnie pozostają w strefie propozycji. Pytanie jakie należy zadać w tym miejscu dotyczy kwestii korzyści, jakie daje użycie grafletów większego rozmiaru. Jak duża poprawa detekcji różnic następuje i gdzie leży kompromis pomiędzy liczbą grafletów a dokładnością miary.

Liczba 151 pn-grafletów pozwala na porównanie wzrostu jakości (detekcji różnic) w przejściu z 4 na 5 wierzchołków, jak i 3 na 4. Przypadek najmniejszy z 2 na 3 zostaje pominięty, ze względu na prostotę podsieci tych rozmiarów.



Rysunek 6.10: Różnice w detekcji różnic za pomocą GDDA w zależności od wzrostu rozmiaru wykorzystanych grafletów.

Korzystając z sieci wygenerowanych w poprzednim teście możemy w prosty sposób sprawdzić zmiany, zmniejszając liczbę interpretowanych wierszy w macierzy GDD do tych odpowiadających orbitom z pn-grafletów o danej wielkości. Wyniki tej operacji zostały przedstawione na trzech diagramach na Rysunku 6.10 odpowiadających kolejno coraz większym zbiorom pn-grafletów.



Rysunek 6.11: Diagramy przedstawiające uśredniony spadek średniej wartości GDDA dla konkretnych rozmiarów sieci wraz ze wzrostem użytych pn-grafletów.

Wzrost jakości detekcji różnic pomiędzy sieciami jest obserwowany we wszystkich testowych przypadkach, nie jest on jednak równomierny. Na Rysunku 6.11 za pomocą diagramu zobrazowano skalę przyrostu jakości detekcji GDDA. Pomiedzy przejściem z 3- na 4-, a 4- na 5-wierzchołkowe pn-graflety, obserwujemy pewne różnice. W pierwszym przypadku (lewy diagram) widoczny jest wzrost detekcji różnic dla sieci z przewagą liczby tranzycji nad liczbą miejsc. W tym miejscu można postawić hipotezę, że jest to spowodowane ograniczeniami, jakie zostały nałożone na testowe sieci (zakaz występowania miejsc wejściowych i wyjściowych) oraz charakter małych pn-grafletów. Odpowiadają one bardzo prostym i nielicznym (8 podsieci) strukturom o rozmiarze do 3-wierzchołków, posiadającym niewielkie możliwości detekcyjne. Dodatkowo, możliwość wystąpienia wyjściowych i wejściowych tranzycji spowodowała wykrycie większej liczby pn-grafletów przy zastosowaniu grafletów o rozmiarze do 4-wierzchołków.

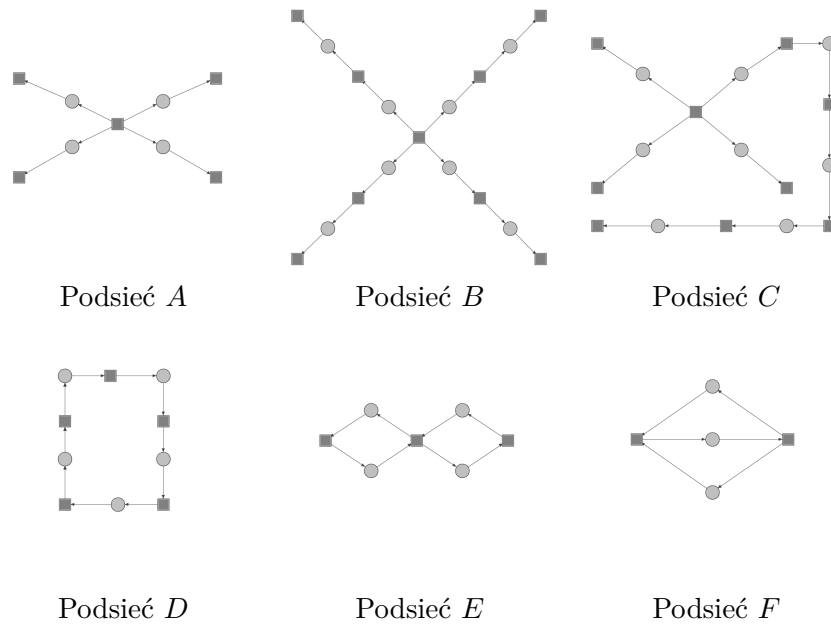
Natomiast w przypadku przejścia z pn-grafletów o rozmiarze 4 na pn-graflety o rozmiarze 5 widoczne jest inne umiejscowienie (na diagramie) wykrytych różnic. Jest to spowodowane możliwością wykrycia nowych, większych grafletów i ich wpływu na sieć. Wraz ze wzrostem rozmiarów sieci obserwowany jest spadek średniej wartości podobieństwa, co odpowiada zwiększonym możliwościom detekcji. Maksymalny obserwowany wpływ na średnią wartość jest mniejszy przy przejściu z pn-grafletów rozmiarów 4 na 5 niż przy przejściu z pn-grafletów rozmiarów 3 na 4, kolejno wynosząc 0,09 jednostki GDDA i 0,14 jednostki GDDA.

Wnioski z przeprowadzonego testu wskazują stopniowy spadek przyrostu jakości w wyniku stosowania coraz większych grafletów. Należy jednak pamiętać, że stosujemy je tutaj do stosunkowo małych sieci o niewielkiej gęstości. Zachowanie GDDA w przypadku sieci PPI ze względu na rozmiar i gęstość może skutkować znacznie większym i stabilniejszym przyrostem niż w badanym przypadku sieci Petriego. Niezmiennym pozostaje fakt, że możliwość użycia większej liczby grafletów wpłynie pozytywnie na możliwości wykrywania różnic. Natomiast dobranie odpowiedniego rozmiaru grafletów będzie nadal zależać od możliwości obliczeniowych współczesnych komputerów, przy czym prawdopodobieństwo możliwości stosowania grafletów o rozmiarach większych niż 8 [49], pozostanie na bardzo długi czas w sferze teoretycznej.

Podsieć ADT jako istotna różnica pomiędzy modelami – GDDA

Interpretacja wyniku GDDA znacząco różni się od interpretacji stopnia podobieństwa wyrażonego przez maksymalny wspólny podgraf, czy dystans grafowy, mimo że na jednym z etapów dystans jest wyznaczany. Jednocześnie, pytania jakie zadawane są przy interpretacji wyników, pozostają niezmiennie, np. czy różnice pomiędzy modelami mają formę spójnego podgrafu, czy rozproszonych po całej sieci pojedynczych wierzchołków, czy jakąś pośrednią formę? Czy za pomocą grafletów jest możliwe ustalenie jednoznacznej odpowiedzi na to pytanie? Niestety nie, co spowodowane jest sposobem działania metody. Możliwe jest jednak wykrycie różnic pomiędzy sieciami i przybliżonego określenia ich formy.

Dla sieci Petriego została już wcześniej zaproponowana definicja struktur odpowiadających biologicznie istotnym podprocesom (zbiory MCT/ADT, a zwłaszcza spójny wariant conADT, zostały dokładnie opisane w Rozdziałach 4 i 9). Korzystając z nich możliwe jest dokonanie symulacji porównania modeli sieci, zawierających istotne z biologicznej perspektywy różnice. Pozwoli to na lepszą interpretację wyników GDDA, określenie skali i w niektórych przypadkach formy, jaką posiadają różnice.



Rysunek 6.12: Struktury indukowane przez nietrywialne zbiory ADT użyte w ramach eksperymentu.

Jako że podsieci oparte o zbiory ADT mogą przedstawiać struktury należące do różnych klas grafów, testowane przypadki powinny je reprezentować. Sześć podsieci reprezentujących podprocesy biologiczne zostało przedstawionych na Rysunku 6.12.

Podsieci A-C reprezentują drzewiaste struktury, które często można zaobserwować w sieciach Petriego bez względu na obszar, w którym się znajdują. Za użyciem w testach tych trzech podobnych struktur stała się chęć sprawdzenia zachowania się GDDA w przypadku niewielkich różnic w podsieciach reprezentujących tę samą klasę (podobnie sytuacja wygląda dla podsieci D-F). Podsumowując relacje pomiędzy nimi: podsieć A zawiera się w obu pozostałych, podsieci B i C posiadają taką samą liczbę wierzchołków o takich samych stopniach (tak więc spełniają minimalny warunek konieczny do zajęcia

izomorfizmu, ale nie są izomorficzne – różnica w wystąpieniach pn-grafletów G_{41}^{PN} i G_{48}^{PN}).

Pojawia się więc pytanie, czy można określić izomorfizm za pomocą grafletów? Odpowiedź brzmi tak, na bazie podobieństwa do metod sprawdzających izomorfizm poprzez znalezienie wszystkich cykli czy ścieżek w grafie. Wpływ na praktyczną aplikację ich do tego celu ma rozmiar użytych grafletów, rozmiar i gęstość sieci. Z tego powodu za pomocą grafletów można potwierdzić izomorfizm dla bardzo małych sieci lub o charakterystycznej budowie.

Struktury D-F odpowiadają podsieciom zawierającym przynajmniej jeden cykl. W ich przypadku różnice polegają na różnych miejscach styku (lub jego braku w przypadku D) pomiędzy cyklami. Efektem będzie występowanie pn-grafletów z rodziny $G_{16}^{PN} - G_{23}^{PN}$ odpowiadających węzłowi w drzewie i występujących w podsieciach A-C, a nieobecnych w strukturze D, będącej pojedynczym cyklem. W momencie połączenia podsieci za pomocą łuku z testową siecią zaobserwujemy powstanie nowych $G_{16}^{PN} - G_{23}^{PN}$, o czym jest mowa w kolejnych paragrafach.

Reakcja GDDA i RGF na podsieć ADT

Przed rozpoczęciem bardziej skomplikowanych testów dotyczących GDDA i jej reakcji na struktury oparte ADT, warto sprawdzić jak ocenia ona podobieństwo pomiędzy opisanymi podsieciami A-F. Każda z podsieci została porównana z pozostałymi, czego rezultat został przedstawiony w Tabeli 6.3.

Tabela 6.5: Rozkład grafletów w podsieciach A–F z Rysunku 6.12

Tabela 6.3: Wyniki porównania podsieci A-F za pomocą PN-GDDA

	A	B	C	D	E	F
Podsieć A	1,000	0,877	0,864	0,797	0,802	0,887
Podsieć B		1,000	0,954	0,840	0,745	0,782
Podsieć C			1,000	0,880	0,742	0,768
Podsieć D				1,000	0,782	0,819
Podsieć E					1,000	0,853
Podsieć F						1,000

Tabela 6.4: Wyniki porównania podsieci A-F za pomocą PN-RGF

	A	B	C	D	E	F
Podsieć A	0	0,665	0,605	1,547	1,377	1,452
Podsieć B		0	0,189	1,140	0,978	1,302
Podsieć C			0	1,011	0,972	1,261
Podsieć D				0	0,984	1,0
Podsieć E					0	0,894
Podsieć F						0

	A	B	C	D	E	F
Gralet G_0^{PN}	4	8	8	5	4	3
Gralet G_1^{PN}	4	8	8	5	4	3
Gralet G_2^{PN}	4	8	8	5	4	3
Gralet G_5^{PN}	0	4	4	5	6	4
Gralet G_6^{PN}	0	0	0	0	1	1
Gralet G_7^{PN}	6	6	6	0	1	1
Gralet G_{13}^{PN}	4	4	4	0	0	0
Gralet G_{14}^{PN}	0	0	0	0	2	1
Gralet G_{15}^{PN}	0	0	0	0	2	1
Gralet G_{16}^{PN}	0	4	4	5	2	0
Gralet G_{17}^{PN}	0	0	0	0	2	0
Gralet G_{20}^{PN}	12	12	12	0	2	0
Gralet G_{21}^{PN}	0	4	4	5	2	0
Gralet G_{24}^{PN}	0	0	0	0	2	2
Gralet G_{30}^{PN}	0	0	0	0	0	1
Gralet G_{31}^{PN}	0	4	4	5	2	0
Gralet G_{34}^{PN}	0	0	0	0	1	0
Gralet G_{37}^{PN}	6	6	6	0	1	0
Gralet G_{41}^{PN}	0	0	3	5	0	0
Gralet G_{48}^{PN}	0	12	3	0	0	0
Gralet G_{64}^{PN}	0	0	0	0	2	0
Gralet G_{67}^{PN}	0	0	0	0	2	0
Gralet G_{68}^{PN}	12	12	12	0	0	0
Gralet G_{80}^{PN}	1	1	1	0	0	0
Gralet G_{82}^{PN}	0	0	0	0	1	0
Gralet G_{85}^{PN}	0	0	0	0	2	0
Gralet G_{86}^{PN}	0	0	0	0	2	0
Gralet G_{125}^{PN}	0	0	0	0	0	1

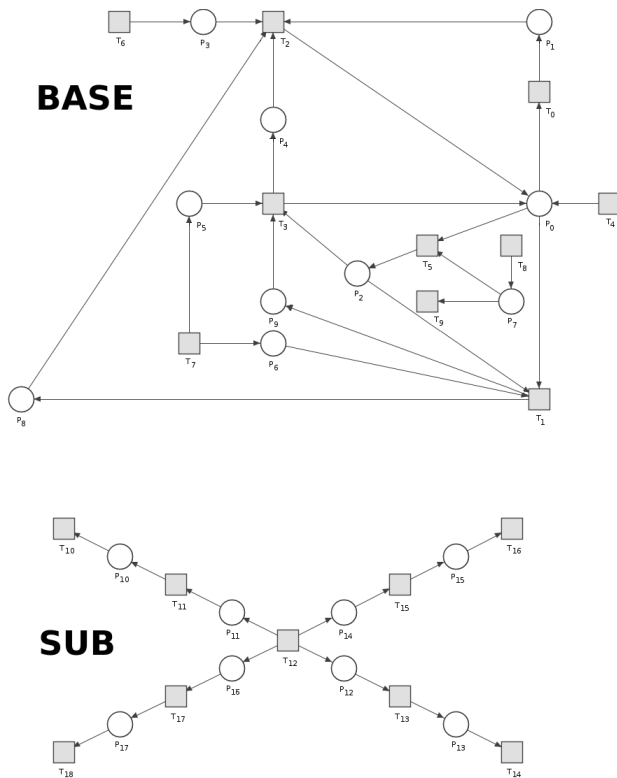
Największą wartość GDDA osiągnęło dla podsieci B i C, co potwierdza obserwacje z poprzedniej podsekcji. W Tabeli 6.5, gdzie przedstawiono rozkład grafletów, różnica pomiędzy podsieciami B i C dotyczy wystąpień G_{41}^{PN} i G_{48}^{PN} , gdy dla pozostałych są one identyczne. Przekłada się to na licznosc orbit zliczaną przez GDDA dla każdego wierzchołka, co łatwo przeanalizować na tych małych modelach. Po przeciwnej stronie jest porównanie podsieci D i E. Podsieć E wykazuje najmniejsze podobieństwa z pozostałymi sieciami, gdzie największe różnice występują w podsieci D. Podobnie jak w opisywanym poprzednio przypadku, różnice widać już z perspektywy wystąpień grafletów.

Analizując wyniki z Tabeli 6.3 można zwrócić uwagę na podsieć D reprezentującą cykl. Mimo zawierania pojedynczego cyklu, podsieć wykazuje większe podobieństwa z podsieciami A-C w rozkładzie grafletów. Podsumowując, bazując na samym GDDA nie jesteśmy w stanie określić jednoznacznie, że dwie porównywane podsieci należą do jednej klasy grafów np. drzew.

Porównując rezultat uzyskany za pomocą GDDA (Tabela 6.3) z wynikami dla RGF (Tabela 6.4) widoczna jest różna reakcja na różnice istniejące pomiędzy pn-grafletami.

Lokalizacja połączeń

Przeprowadzono kolejny test mający ustalić zachowanie GDDA w zależności od tego gdzie występuje połączenie podsieci ADT reprezentującej różnicę. Ma to na celu sprawdzenie czy liczba nowo powstałych grafletów jest na tyle duża, a zróżnicowanie pomiędzy nimi na tyle znaczące, że zostanie wykryte i rozpoznane przez GDDA.



Rysunek 6.13: Przykład dwóch podsieci użytych do sprawdzenia wpływu lokalizacji połączenia (użyty dla testów RGF – Rysunek 6.5).

Tabela 6.6: Porównanie wartości GDDA w zależności od miejsca połączenia podsieci

	BASE	BASE+SUB	Diff
BASE	1	0,793	
BASE+SUB	0,793	1	
t_0^{IN}	0,612	0,733	0,121
t_1^{IN}	0,443	0,554	0,111
t_2^{IN}	0,437	0,554	0,117
t_3^{IN}	0,401	0,515	0,113
t_4^{IN}	0,657	0,785	0,127
t_5^{IN}	0,522	0,644	0,122
t_6^{IN}	0,725	0,863	0,138
t_7^{IN}	0,69	0,822	0,132
t_8^{IN}	0,722	0,857	0,135
t_9^{IN}	0,703	0,863	0,16
t_0^{OUT}	0,59	0,728	0,138
t_1^{OUT}	0,437	0,557	0,12
t_2^{OUT}	0,52	0,646	0,126
t_3^{OUT}	0,427	0,54	0,113
t_4^{OUT}	0,686	0,828	0,142
t_5^{OUT}	0,512	0,646	0,134
t_6^{OUT}	0,738	0,879	0,141
t_7^{OUT}	0,71	0,839	0,128
t_8^{OUT}	0,732	0,874	0,142
t_9^{OUT}	0,674	0,854	0,18

Do testu wykorzystany został fragment większego modelu systemu biologicznego, o rozmiarze 20 wierzchołków, przedstawiony na górze Rysunku 6.13, określany dalej jako BASE. Podsieć rozszerzająca

sieć to znana z poprzedniego testu Podsieć B, dalej określana jako SUB. Rozważane zostały połączenia pomiędzy miejscem p_{11} z podsieci SUB, połączonej pojedynczym łukiem z kolejnymi tranzycjami z sieci BASE, w wyniku których powstawały nowe sieci uczestniczące w tym teście. Jako że połączenie może mieć dwa kierunki, dla każdego z przypadków rozważamy warianty IN (z SUB do BASE) i OUT (z BASE do SUB). W efekcie otrzymujemy sieci identyfikowane przez miejsce połączenia i jego kierunek np., tranzycja t_0^{IN} – odpowiadająca sieci składającej się z BASE i SUB połączonych łukiem zaczynającym się w tranzycji t_0 a kończącym w p_{11} .

Analizując wyniki przedstawione w Tabeli 6.6, można stwierdzić, że miejsce w którym istnieje połączenie z BASE do SUB jest równie istotne co rozmiar różnicy pomiędzy sieciami, co jest spowodowane powstaniem różnych zbiorów $N_i(PN^{Con})$ (wspominanych w kontekście RGF), a następnie ich wpływem na rozkład orbit i GDD. Rozrzut wartości GDDA jest wyraźny, od 0,401 dla połączenia t_3^{IN} , do 0,725 przy połączeniu t_6^{IN} i 0,738 przy połączeniu t_6^{OUT} . Wartość GDDA jest więc silnie zależna od stopnia wierzchołków stycznych do fragmentów sieci występujących jedynie w jednej sieci.

Kolejną obserwacją jest wyraźna różnica w przypadku zmiany kierunku łuku łączącego oba komponenty. Jest to spowodowane powstaniem innego zbioru grafletów, w znaczącym stopniu różniącego się zawartymi grafletami i ich liczbą.

Zwróćmy uwagę na zachowanie GDDA w przypadku porównania z BASE i BASE+SUB, co odpowiada porównaniu sieci BASE z niespójną siecią zawierającą BASE i podsieć SUB, gdzie BASE odpowiada za największy wspólny podgraf. Podobieństwo tych dwóch modeli GDDA określa na 0,793. Pozostałe wyniki z kolumny BASE+SUB z Tabeli 6.6, przedstawiają wpływ na wynik GDDA nowych pn-grafletów i ich orbit powstałych ze względu na połączenie jednym łukiem sieci BASE i SUB.

Interesujące jest zaobserwowanie przypadków porównań spójnych sieci z BASE+SUB, dla których wartość GDDA jest mniejsza niż porównaniu sieci BASE i BASE+SUB. Wpływ grafletów powstałych w wyniku połączenia dwóch sieci z podsiecią i ich orbit na GDD jest tak duży, że uzyskiwana wartość jest zauważalnie mniejsza. Jest to w opozycji do standardowej interpretacji największego wspólnego podgrafu czy dystansu grafowego, gdzie każdy wierzchołek/łuk różnicy jest tak samo ważny. Z perspektywy sieci Petriego wynik GDDA przekazuje nam bardzo pożądaną informację na temat sąsiedztwa różnicy. W przypadkach połączenia z tranzycjami wejściowymi i wyjściowymi, a więc miejscami styku z innymi procesami, obserwujemy oscylowanie w okolicy wartości 0,793, podczas gdy połączenia z łukami o wysokich stopniach znacząco zmienia rozkład wewnątrz sieci. Zaznaczyć trzeba, że wpływ na to mają także sąsiednie wierzchołki i ich stopnie, a nie wyłącznie wierzchołek połączony nowym łukiem.

Porównania różnych rozszerzeń o podsieci conADT

Kolejny test będzie polegał na sprawdzeniu reakcji na różne typy różnic, reprezentowanych za pomocą zbiorów ADT na Rysunku 6.12. Ponownie rozszerzana zostanie sieć z poprzedniego testu (BASE z Rysunku 6.13). Zbiory ADT zostały połączone z tranzycjami t_1 i t_6 , wybranymi ze względu na różne stopnie wierzchołków (t_1 jest stopnia 4 a t_6 stopnia 1) i lokalizację po przeciwnych stronach sieci (wyniki w Tabeli 6.7).

Na podstawie przedstawionych wyników można zaobserwować kilka własności. Potwierdzony jest podobny wpływ na GDDA podsieci drzewiastych B i C, zgodnie z tym co było zaobserwowane w teście wewnętrznego podobieństwa podsieci. Wpływ rozmiaru sieci na rezultat GDDA jest także obserwowal-

Tabela 6.7: Wyniki GDDA przy porównaniu sieci bazowej z siecią rozszerzoną o zadaną podsieć, połączoną tranzycją t_1 lub t_6

	A	B	C	D	E	F
Tranzycja t_1	0,530	0,443	0,440	0,508	0,496	0,546
Tranzycja t_6	0,841	0,725	0,719	0,774	0,740	0,820

ny. Podsieci A i F są najmniejsze rozmiarem w porównaniu do pozostałych rozważanych przypadków i wartości porównań sieci rozszerzonych o nie zwracały największą wartość GDDA co jest zgodne z oczekiwaniami. Są to kolejno 0,530 i 0,546 przy połączeniu podsieci z tranzycją t_1 , oraz 0,841 i 0,820 przy połączeniu z tranzycją t_6 . Wspomniane wcześniej różnice stopni i lokalizacji połączonych tranzycji mają znaczący wpływ na uzyskany rezultat, co było oczekiwane.

Należy teraz spojrzeć na wpływ podsieci ADT z szerszej perspektywy, obserwując zachowanie GDDA na większej próbce sieci. Do tego wykorzystane zostaną testowe przypadki biologicznych sieci wygenerowane podczas testowania średniego podobieństwa. Dla każdej z tych sieci zostanie utworzona nowa sieć rozszerzona o konkretną podsieć ADT. Komponenty sieci zostają połączone pojedynczym łukiem. Porównaniu podlega bazowa sieć ze swoim rozszerzeniem. Następnie ze 100 przykładowych sieci o tym samym rozmiarze i proporcjach liczby tranzycji względem liczby miejsc, wyznaczone są minimalna i maksymalna wartość oraz średnie podobieństwo. Rezultat testu został przedstawiony w postaci diagramów na Rysunku 6.14.

Analizując 6 diagramów z Rysunku 6.14 widać, że Podsieć F ma największy wpływ na średnią wartość GDDA w porównaniu z pozostałymi podsieciami. Stoi to w opozycji do spostrzeżeń z poprzedniego testu, gdzie podsieci A i F , jako najmniejsze, miały nieduży wpływ na wartość GDDA.

Dla przypadków $A - D$ diagramy przyjmują podobną formę i aby lepiej wykryć różnice zachodzące między nimi, wygenerowane zostały diagramy obrazujące relacje pomiędzy każdą parą – Rysunek 6.15. Największe różnice względem pozostałych przypadków obserwowane są dla sieci rozszerzonych o podsieć F . Jednocześnie należy stwierdzić, że największe zmiany są obserwowane dla małych sieci.

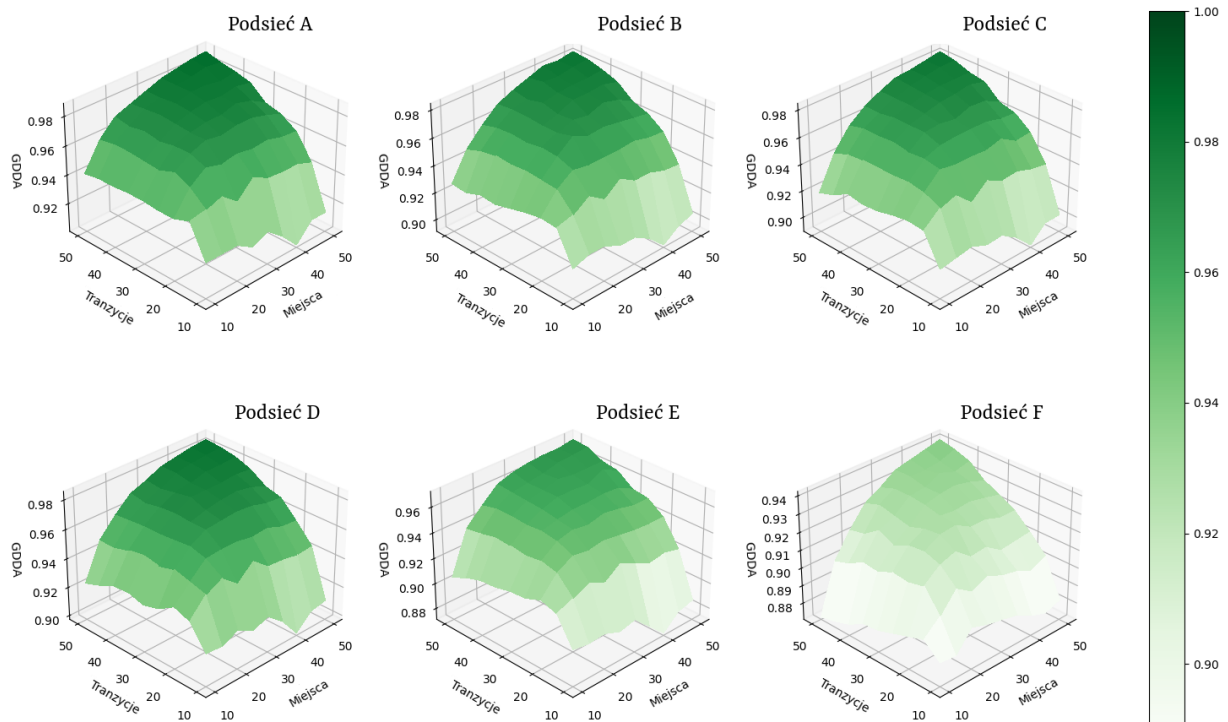
Zwłaszcza na przykładzie F widoczne jest, że dla sieci zawierających 10 tranzycji lub 10 miejsc GDDA cechuje się dużą reakcją na dodaną różnicę i zachodzi to bez względu na liczbę wierzchołków przeciwnego typu.

Analizując wyniki podobieństwa uzyskane dla sieci rozszerzanych przez różne podsieci ADT można zaobserwować, że wykorzystując GDDA nie jest możliwym dokładnie określić formy różnic jaka występuje pomiędzy dwoma porównywanymi sieciami, ale jedynie przybliżoną skalę.

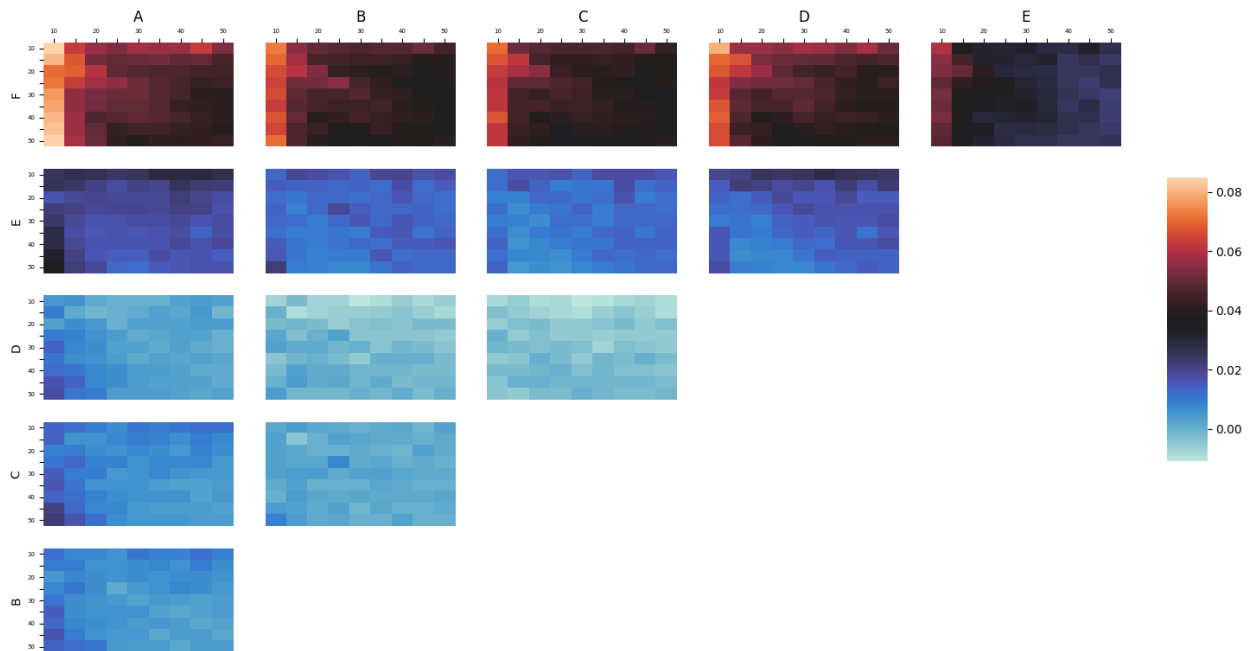
Wielokrotność połączeń

We wszystkich poprzednich testach rozważane były różnice pomiędzy porównywanymi sieciami w formie pojedynczych spójnych podsieci. Konieczne było więc sprawdzenie, jak GDDA reaguje na wielokrotne połączenia i różnice o powtarzającej się strukturze.

Pierwszy test polegał na obserwacji, jak metryka reaguje na wielokrotnie powtarzającą się podsieć. W efekcie do końcowego rozkładu dodany został kilkakrotnie ten sam zbiór grafletów zawierający się w podsieci A . Dodatkowo w wyniku połączenia z bazową siecią, tworzone są graflety połączeniowe, których liczba i rodzaje zależą od miejsca połączenia. Analizując zmiany na diagramach z Rysunku 6.16, które przedstawiają kolejno rozszerzenia o 1, 2 i 3 podsieci A , nie zostały zaobserwowane zmiany

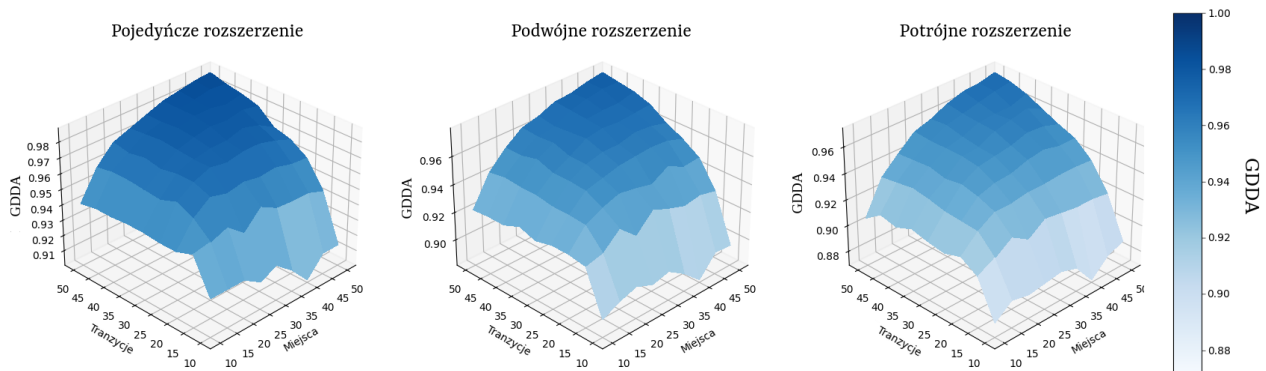


Rysunek 6.14: Diagramy przedstawiające średni wpływ podsieci ADT na testowe sieci.



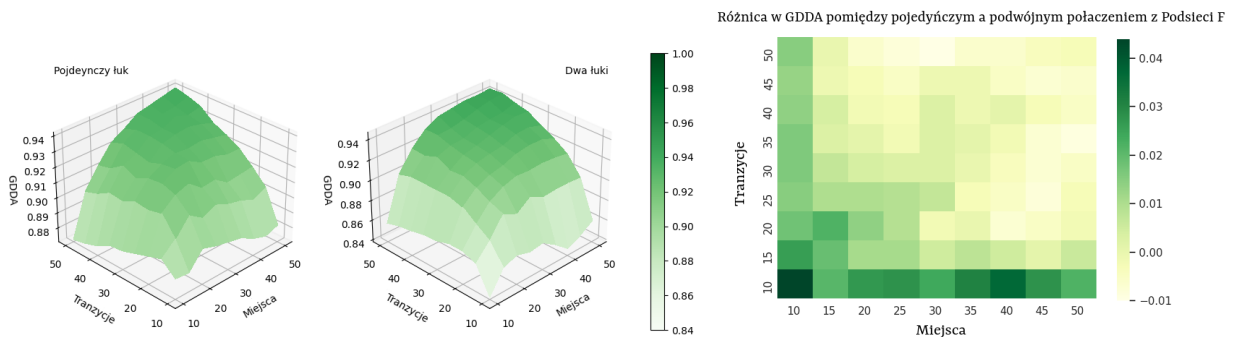
Rysunek 6.15: Diagramy heatmap obrazują różnice w średnim GDDA dla poszczególnych par sieci rozszerzonych o różne podsieci ADT.

w ogólnej charakterystyce przebiegu wartości GDDA. Wraz z pojawieniem się kolejnej różnicy-podsieci widoczny jest spadek wartości GDDA, który jest stały dla sieci o różnych rozmiarach.



Rysunek 6.16: Porównanie zachowania GDDA w przypadku wystąpienia różnicy o postaci jednej, dwóch i trzech podsieci A .

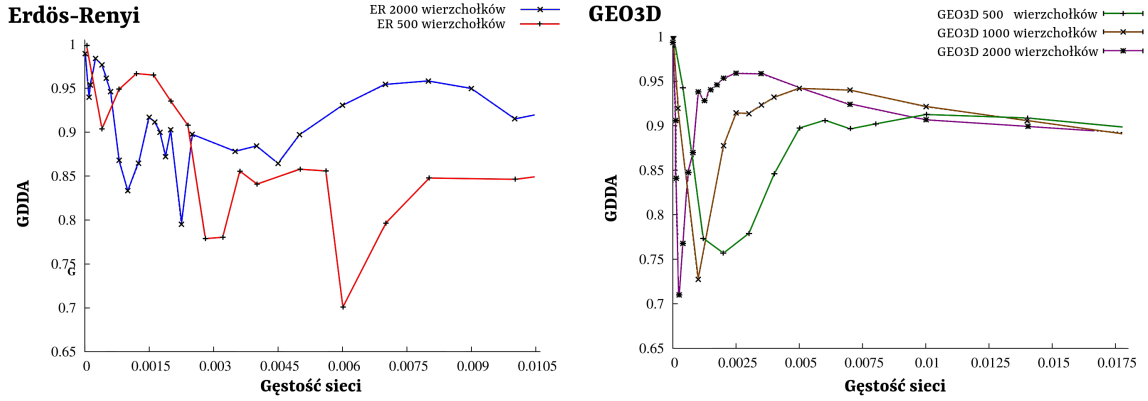
Interesujący jest relatywnie mały wpływ kolejnych takich samych podsieci na GDDA. Jeśli porównamy diagram z różnicą w postaci trzech podsieci A z diagramem dla podsieci F (lewy diagram z Rysunku 6.17), to widać, że podsieć F ma znacznie większy wpływ na rozkład orbit. Dzieje się to pomimo tego, że podsieć F składa się z 5 wierzchołków z 6 łukami, gdy trzy podsieci A odpowiadają 15 wierzchołkom.



Rysunek 6.17: Różnica w rozkładach pn-grafletów pomiędzy podsiecią F połączoną z siecią bazową pojedynczym łukiem i dwoma łukami

Kolejny test polegał na sprawdzeniu zachowania przy użyciu wielokrotnych połączeń pomiędzy podsiecią obrazującą różnicę, a siecią bazową. Liczba nowo powstałych grafletów połączeniowych wynosiła co najmniej tyle samo, co w przypadku połączeń dwóch osobnych podsieci, choć przy małych sieciach lub bliskim sąsiedztwie połączeń zbiór nowo powstałych grafletów może być znacznie większy. W efekcie jest możliwe, że np. pojedyncza sieć połączona przynajmniej dwoma łukami, będzie miała większy wpływ na GDDA niż dwie samodzielne podsieci połączone pojedynczym łukiem.

Na diagramie z Rysunku 6.17, widoczna jest duża różnica dla małych sieci, zwłaszcza dla zbioru wspomnianych wcześniej sieci z liczbą tranzycji lub miejsc równą 10. Wskazuje to na wysoką wrażliwość GDDA w przypadku porównania niewielkich sieci.



Rysunek 6.18: Wykresy przedstawiają zmiany w średnim podobieństwie pomiędzy sieciami PPI wygenerowanymi metodą Erdős-Renyi i GEO3D, przy zmianie gęstości sieci [107].

6.5.3 Stabilność GDDA

W Rozdziale 3 w sekcji poświęconej grafletom został wspomniany problem niestabilności, jaki objawiał się dla części porównywanych sieci. Mowa o przypadkach dla których gęstość mieściła się w przedziale $[0; 0,005]$, co odpowiada niskiej gęstości dla sieci PPI – Rysunek 6.18. Niestabilność ma postać spadku średniej wartości GDDA, która dla części przypadków sięgała 0,2 wartości GDDA. Problem został dokładnie opisany w artykule [107], gdzie po raz pierwszy zaobserwowano i określono niestabilność GDDA, oraz w [50] będącym odpowiedzią na zarzuty postawione metryce GDDA.

Ze względu na wspomnianą dyskusję konieczne okazało się przetestowanie czy niestabilność zachodzi dla pn-grafletów, a jeśli tak, to czy typowe modele biologiczne wyrażone za pomocą sieci Petriego ze względu na swoje niskie gęstości są wrażliwe na niestabilność wartości GDDA.

Aby tego dokonać, został powtórzony eksperyment opisany w artykule [107], po zmianach danych wejściowych na odpowiadające sieciom metabolicznym przedstawionym za pomocą sieci Petriego. Przed przejściem do testów, konieczne było rozwiązanie kwestii definicji gęstości. We wspomnianych oryginalnych przypadkach stosowano definicję dla grafów prostych.

$$\gamma^U = \frac{2|E|}{|V|(|V| - 1)} \quad (6.2)$$

Aby uzyskać lepszą precyzję dla grafów skierowanych, należy użyć wzoru:

$$\gamma^D = \frac{|A|}{|V|(|V| - 1)} \quad (6.3)$$

Po rozszerzeniu o własność dwudzielności, możliwe jest uzyskanie nowego wzoru odpowiadającego sieciom Petriego.

$$\gamma^{PN} = \frac{|F|}{|T||P|}, \quad (6.4)$$

Oryginalny eksperyment z artykułu [107] został przeprowadzony na sieciach ER (Erdős-Renyi), GEO3D (ang. GEO3D – sieci losowe generowane przestrzennie) i ich wariantach – Rysunek 6.18. Każda z sieci posiadała rozmiar równy 500 wierzchołkom. Z perspektywy sieci Petriego modele tych rozmiarów należą do rzadkości, stąd decyzja redukcji do 100 wierzchołków. Podział na tranzycje i miejsca pozostaje w równowadze pół na pół. Mimo że modele biologiczne mają tendencje do nie-

znacznie większej liczby tranzycji niż miejsc (przykłady z Tabeli 6.8), z obserwacji wynika, że zmiany w zachowaniu GDDA nie są znaczące dla rozważanych sieci o rozmiarze 100 (+/- 1 wierzchołek) i dopiero stosunek 1 do 5 wierzchołków różnych typów wykazuje widoczne odchylenia. Na potrzeby przeprowadzenia eksperymentu na pn-grafletach, wygenerowane sieci testowe spełniają następujące charakterystyki:

- rozmiar 100 wierzchołków,
 - 50 miejsc,
 - 50 tranzycji,
- gęstość od 0,039 – 0,119,
- wielokrotne połączenia (ważone łuki),
- brak miejsc wejściowych i wyjściowych,
- tranzycje wejściowe i wyjściowe lub brak obu,
- 100 sieci testowych w ramach jednej gęstości.

Eksperyment polegał na wyznaczeniu średniej wartości GDDA dla 100 sieci o takiej samej gęstości (jest to istotna różnica w porównaniu z poprzednimi testami przeprowadzonymi dla pn-grafletów, gdzie nie brano pod uwagę kwestii gęstości). Każda z sieci jest porównywana z pozostałymi 99. Porównywanie rozpoczynamy od najmniejszej gęstości wynosi 0,046. Następnie zwiększana jest gęstość przez utworzenie 100 nowych sieci o liczbie wierzchołków większej o jeden łuk. W ten sposób możliwe było określenie zakresu gęstości, w którym występuje niestabilność.

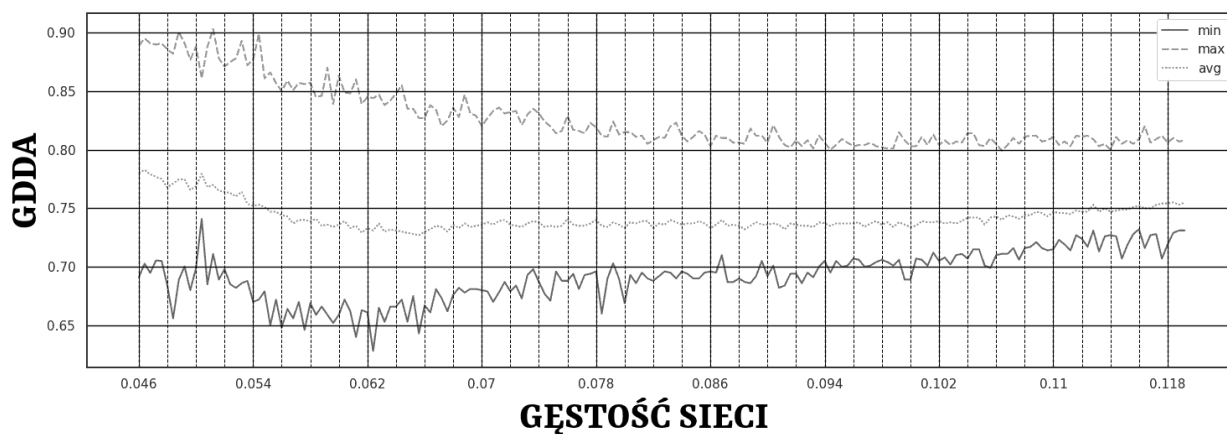
W wyniku przeprowadzonego testu potwierdzone zostało istnienie niestabilności GDDA dla modeli systemów biologicznych wyrażonych w postaci sieci Petriego. Jednocześnie zaobserwowane zostały wyraźne różnice w porównaniu z zachowaniem niestabilności dla nieskierowanych grafów – Rysunki 6.18 i 6.19.

Po pierwsze, spadek średniej wartości GDDA nie następuje natychmiast od najmniej gęstych sieci, jak to ma miejsce w przypadku [107] (Rysunek 6.18), a jest poprzedzony przedziałem dla którego GDDA wykazuje wysoki stopień stabilności.

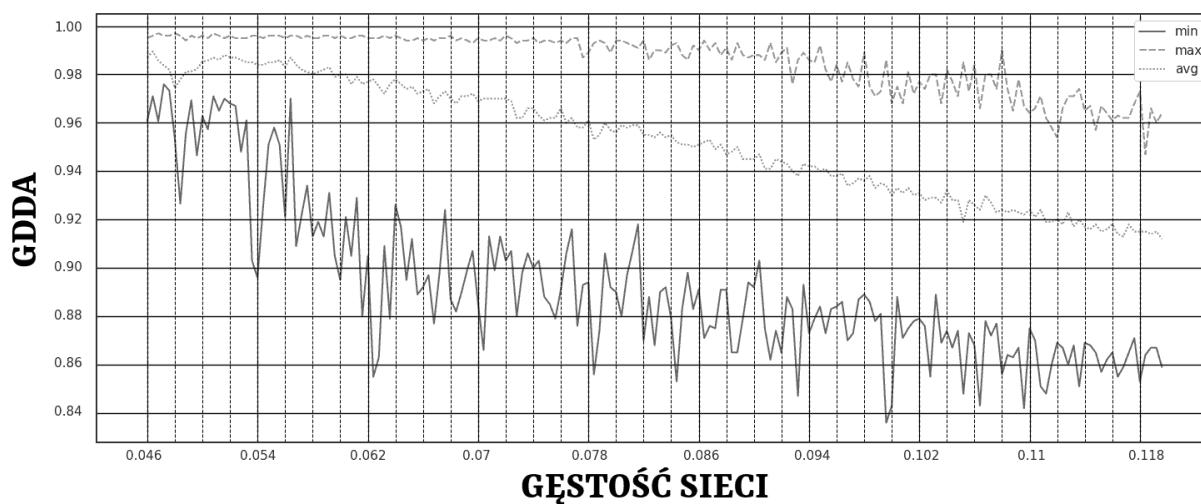
Po drugie, spadek średniej wartości GDDA w porównaniu z tym obserwowanym dla sieci PPI, jest znacząco mniejszy – $\sim 0,2$ odpowiada najniższej zarejestrowanej wartości GDDA dla sieci PPI, a $\sim 0,06$ odpowiada takiemu samemu przypadkowi dla sieci Petriego.

Po trzecie, pomiędzy obszarami odpowiadającymi niestabilności oraz następującym po nim obszarem stabilności, trudno jest określić miejsca w których dochodzi do ustabilizowania GDDA.

W Tabeli 6.8 przedstawiono gęstości dziewięciu modeli systemów biologicznych znalezionych w literaturze. Odpowiadają one różnym rozmiarom sieci, od kilkudziesięciu [51] po prawie czterysta wierzchołków [68]. Z tego powodu nie można bezpośrednio przyrównać przypadków Tabeli 6.8 do rezultatów z Rysunku 6.19 dla sieci o rozmiarze 100 wierzchołków. Jednocześnie możliwe jest stwierdzenie na ich bazie, że modele systemów biologicznych w postaci sieci Petriego pojawiają się na całej przestrzeni badanego spektrum. Część z nich podlega problemowi niestabilności, co nie wyklucza zastosowania dla nich GDDA do porównywania sieci, a wymaga wprowadzenia poprawki w interpretacji uzyskiwanych wyników.



Rysunek 6.19: Wykres przedstawiający średnie, maksymalne i minimalne wartości GDDA uzyskane dla 100 sieci testowych wygenerowanych dla każdej ze sprawdzanych gęstości.



Rysunek 6.20: Wpływ różnicy o formie podsieci opartej o zbiór conADT – rezultat porównania dwóch sieci za pomocą GDDA.

Tabela 6.8: Zbiór sieci Petriego odpowiadających modelom sieci metabolicznych oraz innych systemów biologicznych użytych jako punkt odniesienia podczas testowania niestabilności miary GDDA.

Sieć Petriego	Liczba tranzycji	Liczba miejsc	Liczba łuków	Gęstość	Źródło
K5	45	40	120	0,066	[36]
BER	186	157	557	0,019	[102]
AAA	60	37	124	0,055	[44]
KNS	242	134	575	0,017	[68]
TAK	61	42	127	0,049	[125]
AAH	130	135	294	0,016	[4]
H1	25	26	69	0,106	[51]
H2	28	28	53	0,075	[51]
H3	19	15	52	0,182	[51]

W tym miejscu kończy się powtórzenie eksperymentu z [107], a zaczyna jego rozszerzenie w celu ustalenia reakcji niestabilności GDDA na typowe różnice o formie podsieci conADT. Analogicznie do poprzednich testów, dla każdej z sieci został utworzony dodatkowy model, rozszerzony o podsieć A z Rysunku 6.12. Następnie dokonano porównania oryginalnej sieci z jej rozszerzeniem, wyznaczając minimalną, maksymalną i średnią wartość w ramach jednej gęstości. Rezultat został przedstawiony na Rysunku 6.20. Podobnie jak na Rysunku 6.19 widoczny jest obszar o wysokiej stabilności poprzedzający spadek. Duże wahania wartości GDDA są widoczne zwłaszcza na przebiegu minimalnego podobieństwa. Obszar przejściowy jest także widoczny jedynie dla tego przebiegu.

Podsumowując kwestię niestabilności GDDA w sieciach Petriego, potwierdzony został fakt jej istnienia dla modeli systemów biologicznych, co nie przekreśla jej skutecznego zastosowania. W problematycznym przedziale gęstości będzie się znajdować jedynie część sieci. Jednocześnie w porównaniu z sieciami interakcji międzyproteinowych, wpływ niestabilności na spadek wartości GDDA jest dla nich mniejszy. Dodatkowo dla sieci o bardzo niskiej gęstości GDDA wykazuje wysoką stabilność. To, w połączeniu z tym, że część testowych sieci biologicznych użytych jako referencje plasowała się w tym przedziale, jest argumentem za wykorzystaniem GDDA dla nich. Metryka GDDA jest więc miarodajnym narzędziem do porównywania sieci Petriego, po uprzednim określeniu gęstości sieci i klasyfikacji ich do odpowiedniego przedziału stabilności, co pozwoli na prawidłową interpretację wyniku.

Rozdział 7

Porównanie z użyciem niezmienników i opartych na nich podsieciach

7.1 Wprowadzenie

Rozdział ten poświęcony jest metodzie porównywania sieci Petriego przez dopasowanie t/s-komponentów, powstałej w wyniku modyfikacji algorytmu bazującego na dopasowaniu t-niezmienników opisanego w [10]. Warto zaznaczyć, że metoda z artykułu [10] jest pierwszą opisaną próbą porównywania sieci Petriego i stanowi oczywisty punkt odniesienia dla metod opracowanych w ramach tej dysertacji.

7.1.1 Przyczyny wprowadzonych modyfikacji

Do modyfikacji algorytmu opisanego w artykule [10], przyczyniły się dwie kwestie. Pierwszą jest problemem jego dostępności. Aplikacja CoMeta, w której znajdowała się jego jedyna implementacja, nie jest rozwijana od około 10 lat, a ostatnia wersja zbudowana pod Javę wersji 6 jest niekompatybilna ze współczesnym oprogramowaniem. Jeszcze większy problem występuje ze współpracującą z nią aplikacją MPath2PN [9] do pobierania i transformacji sieci z bazy KEGG [64, 65], która w wyniku braku aktualizacji utraciła możliwość połączenia KEGG, co było jedną z głównych funkcjonalności. Jest to typowy pośród aplikacji naukowych problem opuszczonego i nierozwijanego oprogramowania (ang. abandonware). Opisane problemy z aplikacją były jednym z powodów napisania własnej implementacji wspomnianego wcześniej algorytmu wraz z wprowadzeniem w nim zmian mających na celu polepszenie jakości zwracanych wyników. Wprowadzone modyfikacje bazowały na doświadczeniach i przewidywaniach zespołu profesora Formanowicza działającego w ramach Wydziału Informatyki i Telekomunikacji Politechniki Poznańskiej.

Drugą kwestią jest prostota pierwotnej heurystyki i jej podatność na modyfikacje. Dały one możliwość wprowadzenia szeregu zmian, zaczynając od dostarczanych danych wejściowych, zmiany reguł dopasowywania tranzycji, po analizę dodatkowego typu wierzchołków. Efektem tych zmian jest wzrost możliwości detekcji różnic, a przez to jakości uzyskiwanego porównania.

7.1.2 Modyfikacje

Kwestia identyfikacji tranzycji. Algorytm porównywania przedstawiony w [10] korzystał z bazy (klasyfikacji) Komisyjnych Numerów Enzymów w celu określenia czy dwie tranzycje przedstawiają

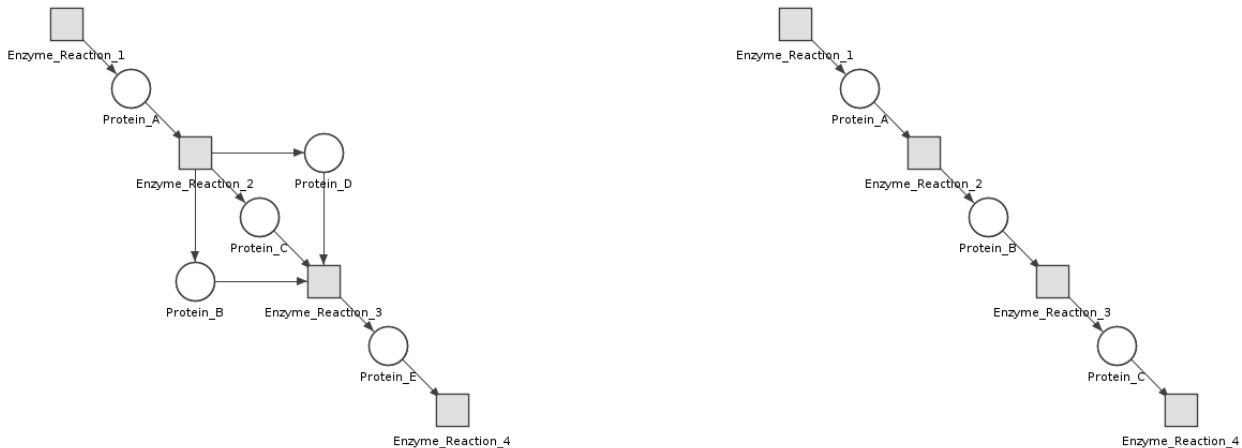
ten sam enzym. Takie rozwiązanie ma zastosowanie ograniczone do sieci kompatybilnych kontekstem z używaną bazą danych. Niestety, dostępność baz danych, które mogłyby być użyte przy identyfikacji wierzchołków w innych obszarach systemów biologicznych, jest często ograniczona. Z tego powodu uniwersalne podejście będzie bazować tylko na danych zawartych w etykietach wierzchołków podczas porównywania modeli.

Jako uniwersalne podejście do identyfikacji użyta zostanie odległość Levenshteina do wyznaczania dystansu pomiędzy dwoma ciągami znaków. Wybór tej metody motywowany jest możliwością zastosowania w jak największej liczbie obszarów naukowych i przemysłowych, pomimo wad jakie to podejście posiada (np. wrażliwość na różne formy zapisu, synonimy). Wykorzystanie odległości Levenshteina pozwala na dokonanie wstępnego dopasowania wierzchołków i stanowi punkt początkowy dla dalszych działań.

Analizując dostępne modele systemów biologicznych przedstawionych za pomocą sieci Petriego, podjęta została decyzja o implementacji dwóch wariantów dopasowania wierzchołków (pomijając wczytanie z zewnętrznego źródła). Dopasowanie zależy od dystansu uzyskanego pomiędzy etykietami:

- idealne (pełne) dopasowanie, gdzie obie etykiety są identyczne i odległość znakowa między nimi jest równa 0,
- akceptowalne dopasowanie, gdzie odległość pomiędzy dwoma etykietami jest równa co najwyżej e , ustalanemu przez użytkownika metody.

Kwestia podobieństwa niezmienników. Określenie kiedy dwa niezmienniki są wystarczająco podobne, aby mogły zostać dopasowane, jest konieczne do efektywnego działania algorytmu. W pierwotnym wariantcie z [10] dla wsparcia niezmiennika z sieci A dopasowywane jest najlepsze (posiadające największą liczbę tranzycji o takich samych nazwach) wsparcie niezmiennika z sieci B , co jest określane z użyciem konkretnego indeksu (Sørensen lub Tanimoto). Na potrzeby części porównywanych przykładów, oprócz najlepszego dopasowania niezmienników rozważany jest wariant idealnego dopasowania. Warto zaznaczyć, że kwestie identyfikacji tranzycji i podobieństwa niezmienników są od siebie zależne. Dla części przypadków porównywania, zastosowanie restrykcyjnej identyfikacji etykiet tranzycji przy jednoczesnym stosowaniu najlepszego dopasowania niezmienników, może dać wynik porównywalny z akceptacją pewnej skali różnic pomiędzy etykietami tranzycji, przy zastosowaniu tylko idealnego dopasowania niezmienników. Kolejnym elementem, który należy rozważyć, jest istnienie sur- i sub- niezmienników, często pomijanych przy analizach, które skupiają się na "prawidłowych" niezmiennikach. Ich wpływ na rezultat porównania został przedstawiony w przykładzie z podrozdziału 7.2.2.



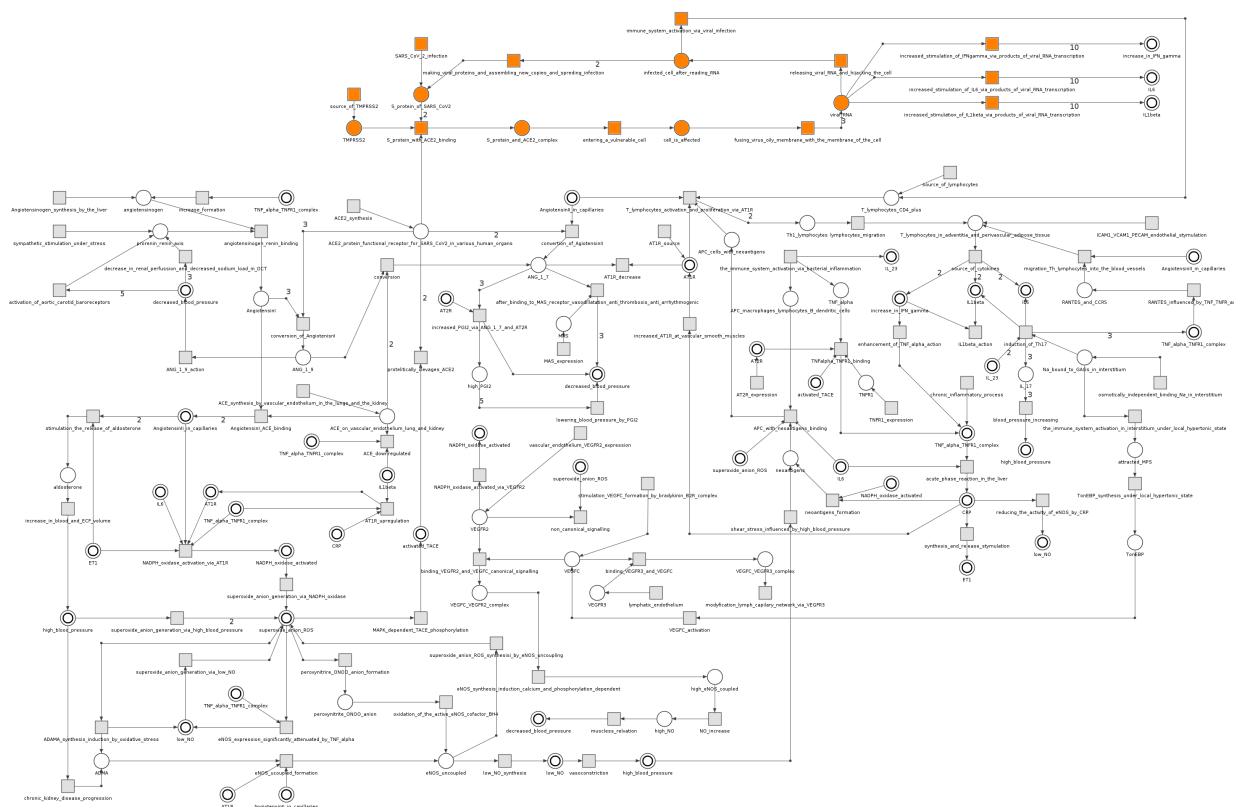
Rysunek 7.1: Przykład przedstawiający dwie podsieci t-komponent pokryte przez jeden t-niezmiennik każda. W wyniku porównania z użyciem algorytmu z [10] oba t-niezmienniki zostaną uznane za identyczne i dopasowane do siebie. Nie zostanie wykryta różnica w postaci trzech miejsc przepływowych, występujących jedynie w podsieci po lewej stronie. Różnica zostanie wykryta jeśli porównaniu podlegają będą całe podsieci odpowiadające t-komponentom, a nie tylko wektory liczb odpowiadające t-niezmiennikom wraz z jego wsparciem.

Przejście z t-niezmienników na t-komponenty Istotną zmianą w stosunku do pierwotnej metody jest przejście z porównywania wsparcia t-niezmienników na porównanie t-komponentów. W efekcie w procesie porównania biorą udział wszystkie wierzchołki, nie tylko należące do jednego typu. Ze względu na charakterystykę strukturalną w t-komponentach występują w zdecydowanej większości miejsca przepływowe. Z tego powodu dla części przypadków mogą zostać pominięte bez straty informacji, czego przykładem jest oryginalny algorytm bazujący na tranzycjach ze wsparcia t-niezmienników. Wadą takiego podejścia jest brak możliwości wykrycia różnic występujących pomiędzy miejscami: ich liczbą i etykietami. Przykład przedstawiony na Rysunku 7.1 przedstawia przykładową różnicę strukturalną pomijaną w oryginalnym wariantcie metody.

7.2 Przykłady

7.2.1 Przykład I – porównanie jako rozszerzenie analizy wyłączania podsieci

Przedstawiony tutaj przykład porównania został zaczerpnięty z pracy [35]. Opisana w tym rozdziale metoda została użyta do poszerzenia istniejącej analizy wyłączania podsieci. Polega ona na wyłączeniu z działania pewnego zbioru wierzchołków w sieci Petriego w celu obserwacji zmian w formie i liczbie t-niezmienników. W porównaniu wykorzystano główny model przedstawiający układ krwionośny pacjenta z szeregiem chorób, w tym infekcją SARS-CoV-2, określany dalej jako **model oryginalny** oraz z drugim modelem będący modelem oryginalnym z wyłączonym (ang. knockout) fragmentem sieci odpowiadającym infekcji SARS-CoV-2. Oba modele przedstawione są na Rysunku 7.2. Ze względu na to, że operacja porównania zachodzi pomiędzy siecią a jej podsiecią, problem identyfikacji tranzycji nie występuje w tym przypadku. W wyniku przeprowadzonego wyłączenia nie powstały żadne niestabilne sub-/sur-niezmienniki, a obie sieci pozostają pokryte przez poprawne niezmienniki. Pomiedzy t-komponentami występować będą tylko idealne dopasowania.



Rysunek 7.2: Sieć przedstawiająca pacjenta z nadciśnieniem, chorobą nerek i infekcją SARS-CoV-2 [35]. Na sieci kolorem pomarańczowym został zaznaczony fragment odpowiadający infekcji.

Interesująca jest różnica w liczbie t-niezmienników pomiędzy sieciami. Oryginalna sieć posiada ich 139451, podczas gdy sieć z wyłączoną infekcją 25997 (dokładne porównanie znajduje się w Tabeli 7.1). Wyłączeniu uległo 11 tranzycji i pośrednio 6 miejsc co daje sumaryczne tylko około 12% całej sieci, jednak ubytek ten odpowiada za 5,3-krotny spadek liczby t-niezmienników. Zaznaczyć należy, że wszystkie t-niezmienniki z sieci z wyłączoną infekcją występują w sieci oryginalnej.

Tabela 7.1: Liczby niezmienników znajdujących się w porównywanych sieciach. Sieć I odpowiada całej sieci przedstawionej na Rysunku 7.2, podczas gdy Sieć II odpowiada modelowi po wyłączeniu fragmentu odpowiadającego za infekcję.

Liczba t-niezmienników w sieci I	139,451
Liczba t-niezmienników w sieci II	25,997
Liczba wspólnych niezmienników	25,997
Liczba t-niezmienników tylko w sieci I	113454
Liczba t-niezmienników tylko w sieci II	0

Przypadek opisywany w tej sekcji przedstawia problematyczną z perspektywy analizy sytuację, w której liczba t-niezmienników przekracza dziesiątki tysięcy i nierealistyczna jest dokładna weryfikacja każdego z nich. Warto w tym miejscu zaznaczyć, że na potrzeby przeprowadzanych analiz powstało kilka wariantów modelu pacjenta z kombinacją opisanych chorób. Jednak spośród nich tylko dla przypadku przedstawionego na Rysunku 7.2 możliwe było wyznaczenie zbioru minimalnych t-niezmienników. Dla pozostałych wariantów posiadających pewne różnice strukturalne, ale nadal współdzielących zdecydowaną większość modelu, wyznaczenie tego zbioru było niemożliwe ze względu na wykładniczy wzrost liczby cząstkowych wektorów. Oczwistym wnioskiem jest, że wspomniane warianty mode-

Tabela 7.2: Zbiór t-niezmienników występujących jednocześnie w oryginalnym i modelu z wyłączoną tranzycją t_{14} , a przez to reprezentujących pełne mapowanie.

Niezmiennik-1	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	1	1	1	0	0	1	0	0	0
Niezmiennik-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0
Niezmiennik-3	1	1	0	0	0	0	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0
Niezmiennik-4	3	3	0	0	0	0	0	2	0	3	0	1	2	2	0	0	1	1	0	0	0	0	0	0
Niezmiennik-5	1	1	2	0	0	0	0	2	0	1	0	1	0	0	0	0	1	1	2	0	0	0	0	2
Niezmiennik-6	3	1	0	0	0	0	0	2	0	1	0	1	0	0	0	0	1	1	2	0	0	0	2	0
Niezmiennik-7	0	0	0	1	0	0	0	0	1	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
Niezmiennik-8	0	0	0	3	0	0	0	2	0	0	3	1	2	2	0	0	1	1	0	0	0	0	0	0
Niezmiennik-9	0	0	2	1	0	0	0	2	0	0	1	1	0	0	0	0	1	1	2	0	0	0	0	2
Niezmiennik-10	2	0	0	1	0	0	0	2	0	0	1	1	0	0	0	0	1	1	2	0	0	0	2	0
Niezmiennik-11	0	0	0	0	0	0	0	1	1	0	1	0	1	0	0	0	0	1	0	0	1	1	0	0
Niezmiennik-12	0	0	0	0	0	0	0	5	0	0	3	1	2	2	0	0	1	4	0	0	3	3	0	0
Niezmiennik-13	0	0	2	0	0	0	0	3	0	0	1	1	0	0	0	0	1	2	2	0	1	1	0	2
Niezmiennik-14	2	0	0	0	0	0	0	3	0	0	1	1	0	0	0	0	1	2	2	0	1	1	2	0

z nich nie posiadał w swoim wsparciu wyłączonej tranzykcji t_{14} (odpowiadająca jej kolumna zaznaczona jest kolorem czerwonym), więc występuje dla obu wariantów sieci. Powyższe przypadki odpowiadają pełnym mapowaniom t-niezmienników, gdzie dwa inwarianty o wsparciach uznanych identyczne istnieją w obu porównywanych modelach.

Tabela 7.3: Przykład zawierania się powstałego w wyniku wyłączenia sur-t-niezmiennika w Niezmienniku-15 z oryginalnej sieci.

Sur-niezmiennik-1	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	1	0	0	1	0	0	0
Niezmiennik-15	0	0	0	0	0	1	1	1	1	0	0	0	1	0	1	0	0	1	0	0	1	0	0	0

Tylko jeden sur-t-niezmiennik powstały w wyniku wyłączenia tranzykcji t_{14} może być łatwo powiązany z nieobecnym niezmiennikiem z oryginalnej sieci. Pomiedzy *Sur-niezmiennikiem-1* a *Niezmiennikiem-15* zachodzi relacja zawierania (Tabela 7.3).

Tabela 7.4: Zbiór trzech sub-t-niezmienników występujących w modelu z wyłączoną tranzycją t_{14} nie zawierających się w innych t-niezmiennikach z oryginalnego modelu.

Sub-niezmiennik-2	1	1	0	0	2	0	2	2	0	1	0	1	0	0	0	0	1	1	0	0	0	0	0	0
Sub-niezmiennik-3	0	0	0	1	2	0	2	2	0	0	1	1	0	0	0	0	1	1	0	0	0	0	0	0
Sub-niezmiennik-4	0	0	0	0	2	0	2	3	0	0	1	1	0	0	0	0	1	2	0	0	1	1	0	0

Trzy pozostałe sur-t-niezmienniki nie zawierają się w żadnym z pozostałych poprawnych t-niezmienników i reprezentują nowe niestabilne przepływy powstałe na skutek wyłączenia tranzykcji t_{14} (Tabela 7.4).

Tabela 7.5: Zbiór t-niezmienników występujących jedynie w oryginalnym modelu i niezmapowanych z t-niezmiennikami występującymi w modelu o wyłączonej tranzykcji t_{14} .

Niezmiennik-16	0	0	0	0	0	3	3	5	0	0	0	1	2	2	3	0	1	4	0	0	3	0	0	0
Niezmiennik-17	0	0	2	0	0	1	1	3	0	0	0	1	0	0	1	0	1	2	2	0	1	0	0	2
Niezmiennik-18	2	0	0	0	0	1	1	3	0	0	0	1	0	0	1	0	1	2	2	0	1	0	2	0
Niezmiennik-19	0	0	0	0	2	0	2	2	1	0	0	1	1	0	2	0	1	1	0	0	0	0	0	0
Niezmiennik-20	0	0	0	0	3	0	3	4	0	0	0	2	1	1	3	0	2	2	0	0	0	0	0	0
Niezmiennik-21	0	0	1	0	1	0	1	2	0	0	0	1	0	0	1	0	1	1	1	0	0	0	0	1
Niezmiennik-22	1	0	0	0	1	0	1	2	0	0	0	1	0	0	1	0	1	1	1	0	0	0	1	0

Natomiast t-niezmienniki od numeru 16 do 22 zachodzą tylko w oryginalnej sieci i nie posiadają wyraźnych powiązań z nowo powstałymi sur-t-niezmiennikami z sieci o wyłączonej tranzykcji (Tabela 7.5).

W obu wariantach, oryginalnym i z wyłączoną tranzycją, nie występują poprawne p-niezmienniki, a jedynie 4 non-p-niezmienniki. Non-p-niezmiennik reprezentuje podproces, niebędący poprawnym p-niezmiennikiem, a wektor powstały w wyniku mnożenia y z A zawiera jednocześnie przynajmniej jedną liczbę dodatnią i przynajmniej jedną liczbę ujemną. Wyłączenie tranzycji t_{14} nie wpływa w żaden sposób na ich występowanie, ponieważ jest ona tranzycją wyjściową i nie występuje ani we wsparciu p-niezmienników ani w odpowiadających im s-komponentach. Dlatego operacje porównania zostaną przeprowadzone wyłącznie na t-niezmiennikach. W opisywanym przypadku, podobnie jak w poprzednim z sekcji 7.2.1, przejście z t-niezmienników na t-komponenty nie wpływa na końcowy wynik porównania. W przypadku gdyby wyłączeniu podlegało miejsce p pokryte przez wsparcie przynajmniej jednego p-niezmiennika, nastąpiłaby oczywiście zmiana rozmiaru i możliwa byłaby zajście niestabilnego sub-p-niezmiennika. Korzystając z metody porównywania opartej o niezmienniki, stopień podobieństwa oparty na przepływach wewnątrz modelu został przedstawiony w Tabeli 7.6.

Tabela 7.6: Wyniki porównania z użyciem t-niezmienników, z uwzględnieniem sur-t-niezmienników i bez.

Wyłączone tranzycje	Wynik dla poprawnych t-niezmienników	Wynik dla poprawnych i sur-t-niezmienników
NRF2_inhibition - T_{14}	0,63	0,66
reduction_of_ROS_level - T_{15}	0,95	0,95
T_{14} & T_{15}	0,59	0,619

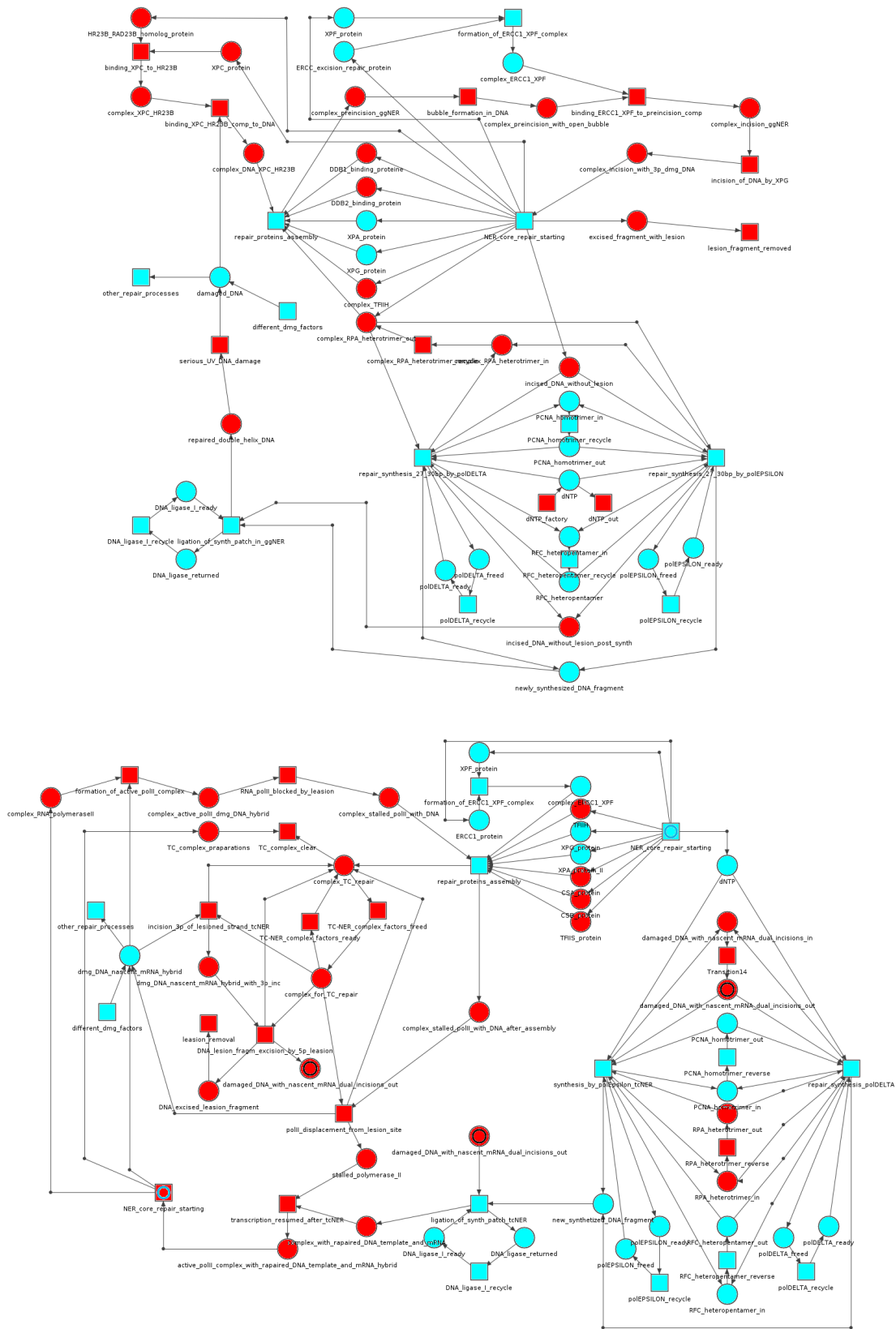
Należy zaznaczyć, że nie każde usunięcie tranzycji wprowadzi system w niestabilność, np. wyłączenie wyjściowej t_{15} *NRF2_inhibition* (kolor niebieski) spowoduje usunięcie Niezmiennika-1, bez utworzenia dodatkowych sur-niezmienników. Wpływ wyłączenia tranzycji t_{14} i t_{15} na stopień podobieństwa niezmienników został przedstawiony w Tabeli 7.6.

7.2.3 Przykład III – mapowanie etykiet i niezmienników

Poprzednie dwa przykłady przedstawiały niestandardowe użycie metody porównywania jako rozszerzenia analizy wyłączania tranzycji. Dzięki operowaniu na tej samej sieci i jej podsięci kwestia dopasowania etykiet wierzchołków była trywialna. Natomiast w tym podrozdziale przedstawione zostanie porównanie dwóch modeli przedstawiających dwa rodzaje procesów naprawy DNA przez wycinanie nukleotydów przedstawionych na Rysunku 7.4.

Dopasowanie etykiet. Obie z porównywanych sieci zostały zbudowane przez jednego autora w krótkim odstępie czasowym, co daje przesłanki do unifikacji stylu i opisu znajdującego się w etykietach. Pełna unifikacja nie została przeprowadzona, co widać w uzyskanych wynikach.

W procesie sprawdzania etykiet idealne dopasowanie znaleziono dla 22 wierzchołków – 8 tranzycji i 14 miejsc. Przy zastosowaniu odległości Levenshteina z dopuszczeniem dystansu równego 4, znaleziono dodatkowe dopasowania 3 tranzycji i 1 miejsca, których poprawność została potwierdzona wiedzą ekspercką. Końcowym etapem było sprawdzenie pozostałych wierzchołków ręcznie, w wyniku czego jedna dodatkowa tranzycja została powiązana ze swoim odpowiednikiem w drugiej sieci. Efektem końcowym jest dopasowanie 31 wierzchołków – 13 tranzycji i 18 miejsc – co stanowi około 57% wierzchołków z sieci GG-NER i około 52% sieci TC-NER – Rysunek 7.4. Jeśli zgodnie z oryginalnym algorytmem

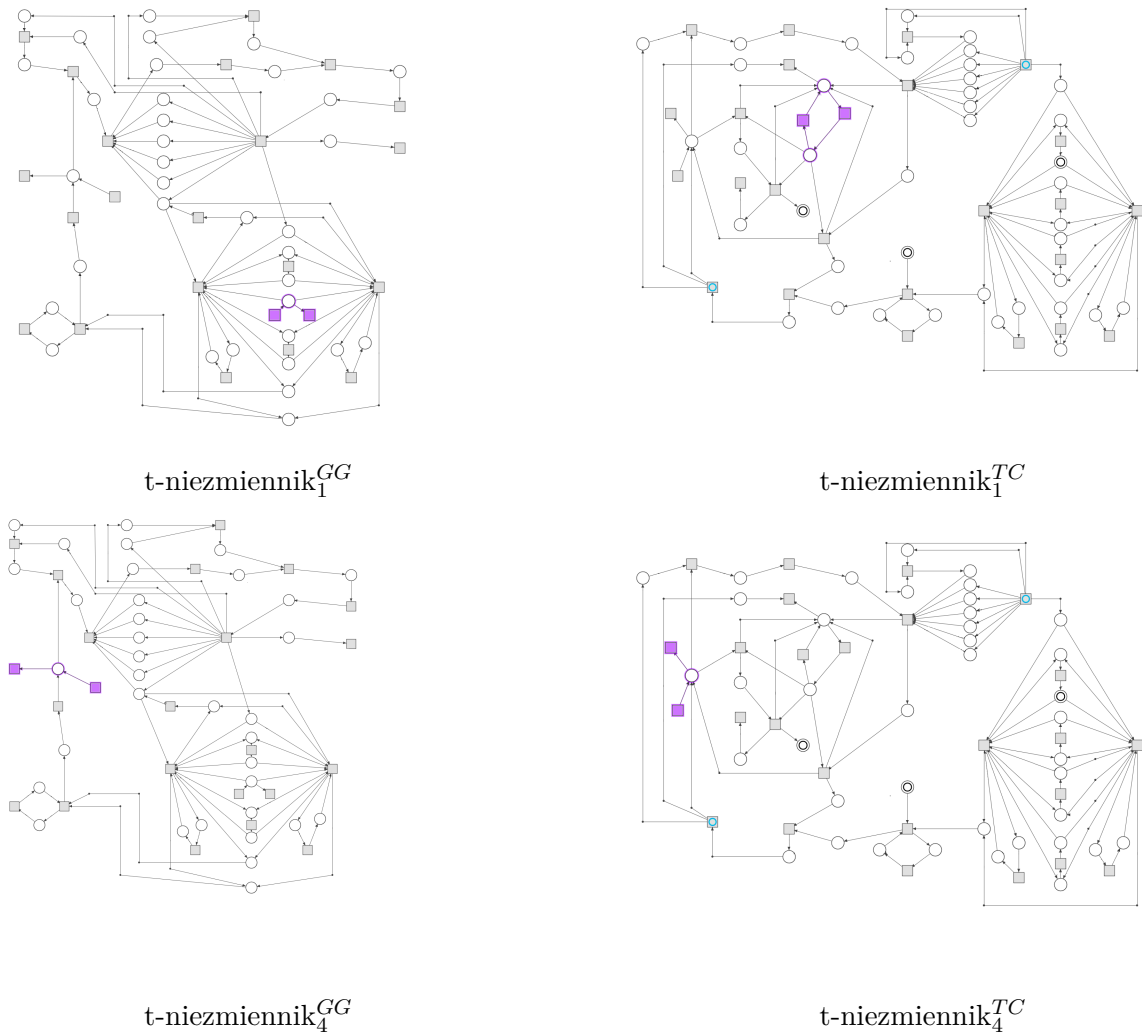


Rysunek 7.4: Znalezione dopasowania pomiędzy etykietami wierzchołków w sieciach GG-NER (górną) i TC-NER (dolną). Kolorem niebieskim zaznaczono wierzchołki z etykietami występującymi w obu modelach, a czerwonym te, których etykiety nie mają odpowiednika w drugiej sieci.

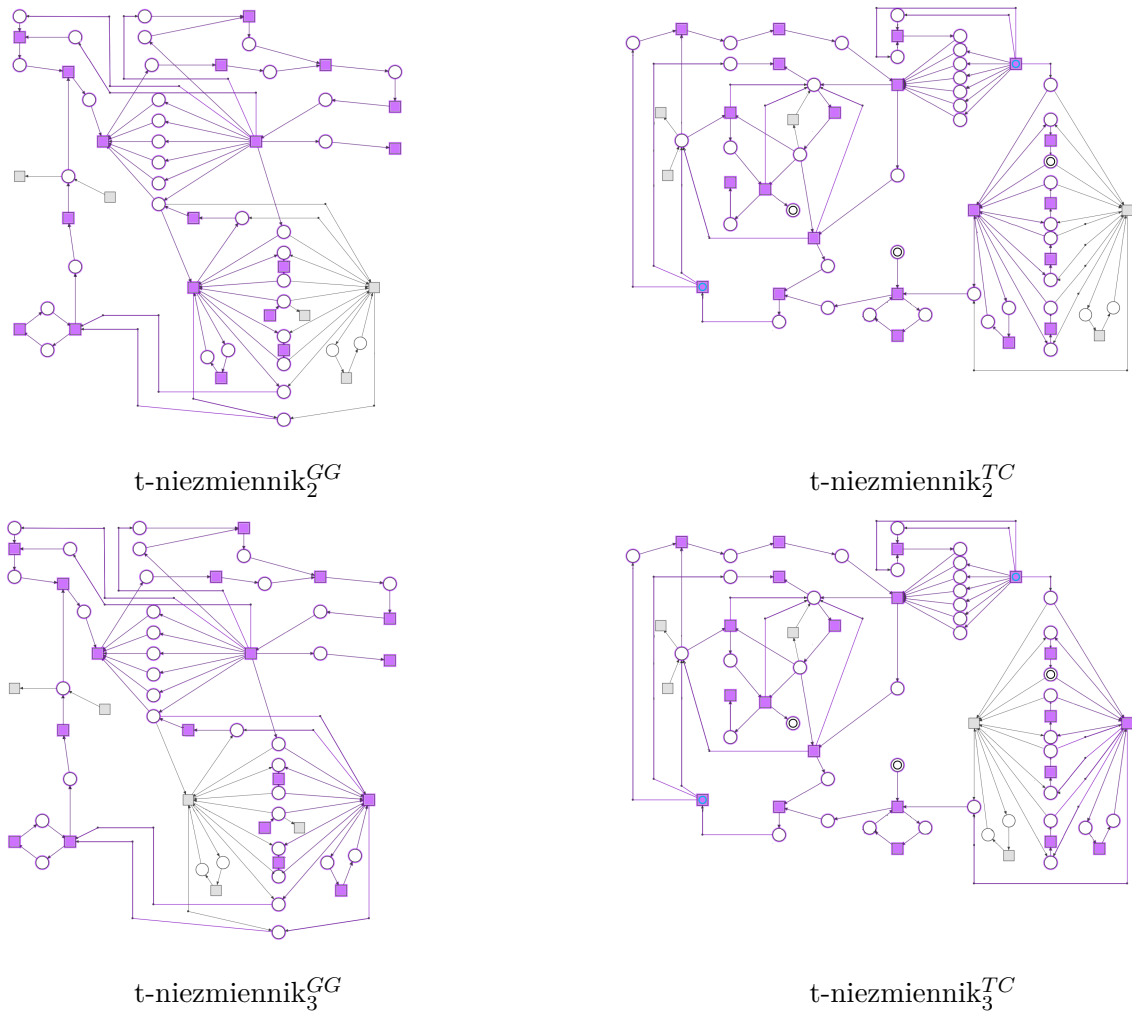
[10] ograniczymy się tylko do tranzycji, to wartości procentowe wynoszą około 53% tranzycji sieci GG-NER, około 49% tranzycji sieci TC-NER.

Porównanie t-niezmenników i t-komponentów. W obu sieciach występują po 4 t-niezmenniki, których t-komponenty zostały przedstawione na Rysunkach 7.5 i 7.6. Idealne dopasowanie występuje dla trywialnych dwuelementowych niezmienników $t\text{-niezmennik}_4^{GG}$ i $t\text{-niezmennik}_4^{TC}$, co stanowi 25% wszystkich t-niezmenników i ustawia wartość podobieństwa na 0,25 (Sørensen), czy 0,14 (Tanimoto). Rozmiar dopasowanych niezmienników nie ma wpływu na wartość podobieństwa ustalaną przez metodę, co może powodować błędną interpretację wyniku.

W przypadku, gdy dopuszczone zostanie dopasowanie częściowe (domyślne podejście), znalezione zostają dwa dodatkowe dopasowania pomiędzy $t\text{-niezmennikiem}_2^{GG}$, $t\text{-niezmennikiem}_3^{TC}$ i $t\text{-niezmennikiem}_3^{TC}$, $t\text{-niezmennikiem}_3^{TC}$ (na Rysunkach 7.7 i 7.8 przedstawiono lokalizacje zmapowanych wierzchołków (dopasowanych na podstawie etykiet) w ramach t-niezmenników/t-komponentów). Poziom podobieństwa pomiędzy każdą z par niezmienników wynosi 0,47 wedle indeksu Sørensen i 0,314 wedle indeksu Tanimoto. To daje końcowy wynik podobieństwa 0,485 (Sørensen) lub 0,405 (Tanimoto).



Rysunek 7.5: Przykłady trywialnych podsieci indukowanych przez niezmienniki występujące w sieciach GG-NER i TC-NER.



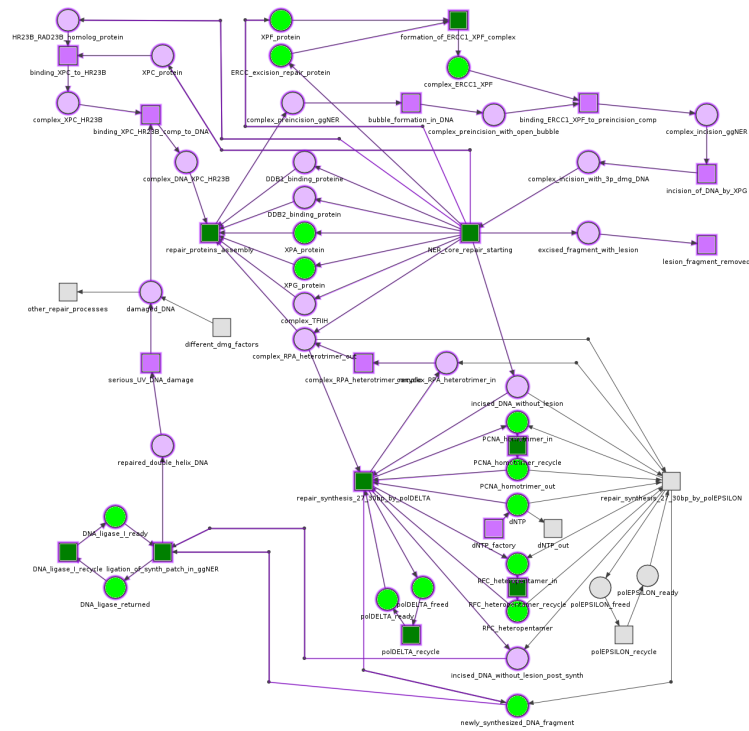
Rysunek 7.6: Przykłady skomplikowanych podsieci indukowanych przez niezmienniki występujące w sieciach GG-NER i TC-NER.

Jako uzupełnienie porównania z użyciem t-niezmienników, zostało przeprowadzone porównanie z wykorzystaniem wszystkich wierzchołków znajdujących się w badanych t-komponentach. W przypadku idealnego dopasowania nie nastąpiła żadna istotna zmiana (dopasowane są trzy wierzchołki w miejsce poprzednich dwóch) – Rysunki 7.7 i 7.8. W przypadku najlepszych dopasowań poziom podobieństwa wynosi 0,448 wg. miary Sørensen i 0,289 wg. miary Tanimoto. Sumaryczne podobieństwo dla danych indeksów wynosi 0,474 i 0,394. W wyniku uwzględnienia wszystkich wierzchołków występujących w ramach t-komponentu, wartość podobieństwa nieznacznie spadła w porównaniu z wynikami uzyskanymi dla t-niezmienników.

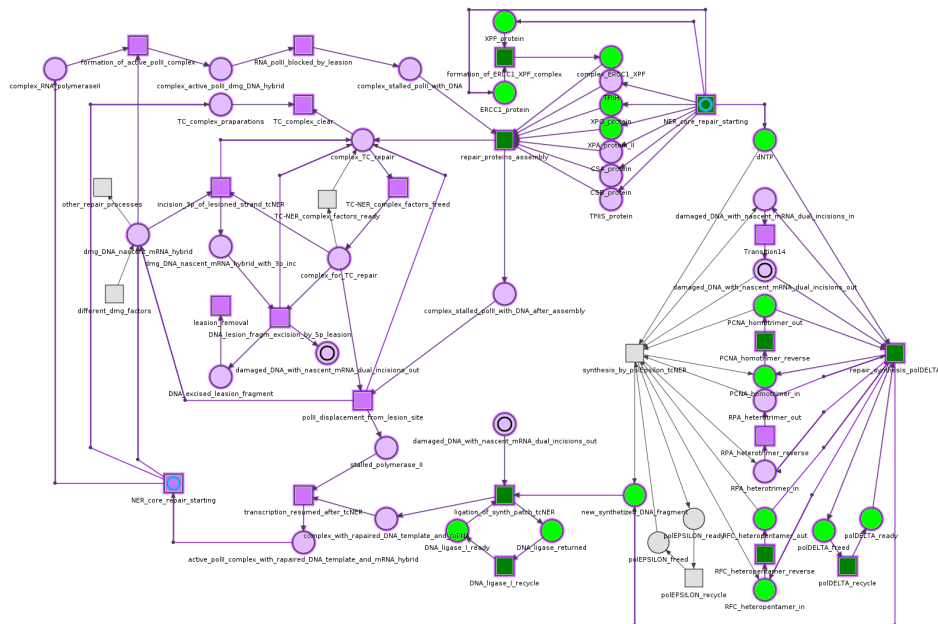
7.3 Podsumowanie

Metoda porównania sieci Petriego z użyciem t/p-niezmienników i t/s-komponentów jest interesującym podejściem ze względu na analizę przepływów zachodzących w sieciach. Tym wyróżnia się od pozostałych metod porównywania opisanych w tej pracy, które opierają się na strukturalnym podobieństwie.

Należy zaznaczyć, że opisana metoda w postaci oryginalnej i po proponowanych modyfikacjach nie pozwala na określenie izomorfizmu pomiędzy dwoma porównywanymi modelami. Jednocześnie



Rysunek 7.7: Stopień zmapowanych wierzchołków w ramach t-komponentu bazującego na inv_2 z sieci GG-NER. Zielonym kolorem zaznaczono miejsca i tranzycje które zostały zmapowane z wierzchołkami w sieci TC-NER, a różowym te które nie znalazły odpowiedników.



Rysunek 7.8: Stopień zmapowanych wierzchołków w ramach t-komponentu bazującego na inv_3 z sieci TC-NER. Zielonym kolorem zaznaczono miejsca i tranzycje które zostały zmapowane z wierzchołkami w sieci GG-NER, a różowym te które nie znalazły odpowiedników.

podobnie do metod sprawdzających spektrum cykli [85] czy zliczających drzewa rozpinające [66], może ona posłużyć do określenia prawdopodobieństwa zachodzenia izomorfizmu.

Istotnym ograniczeniem opisanej metody jest jej zależność od niezmienników i podatność na problem wykładniczego wzrostu liczby cząstkowych wektorów mogący się pojawić przy ich wyznaczaniu. Prowadzi to do dwóch problematycznych sytuacji. W pierwszym przypadku mimo wykładniczego wzrostu liczby cząstkowych wektorów, udało się wyznaczyć zbiór minimalnych niezmienników (ich duża liczba może stanowić wyzwanie przy dalszej analizie, jednak jest ona nadal możliwa). Natomiast w drugim przypadku przyrost cząstkowych wektorów uniemożliwia wyznaczenie zbioru minimalnych niezmienników i blokuje użycie metody porównania opartej o nie.

Kolejnym problemem jest identyfikacja wierzchołków na podstawie etykiet. Stanowi to wyzwanie, zwłaszcza dla przypadków porównywania sieci stworzonych przez różnych autorów (bazujących na innych źródłach). W skrajnych przypadkach możliwe jest, że dwa dopasowane wsparcia niezmienników, mimo posiadania dopasowania etykiet wszystkich wierzchołków, przedstawiają odmienne funkcje w ramach podsieci.

Ostatni problem dotyczy funkcji oceny podobieństwa, która bazuje na liczbie dopasowanych niezmienników, bez uwzględniania ich rozmiaru. Negatywnym skutkiem takiego podejścia może być mylna interpretacja stopnia podobieństwa przy porównywaniu sieci o dużej liczbie dopasowanych trywialnych niezmienników. Przedstawione w sekcji 7.2.3 pojedyncze dopasowanie trywialnych niezmienników t -niezmiennik $_4^{GG}$ i t -niezmiennik $_4^{TC}$ odpowiadało za wartość indeksu Sørenseną równą 0,25, która nie odpowiadała 25% poziomowi podobieństwa porównywanych sieci.

Przejsie z porównania tranzycji na wierzchołki nie zmienia znacząco działania algorytmu, zwiększa jednak istotnie możliwości detekcyjne.

Rozdział 8

Porównanie z użyciem wierzchołków rozgałęziających

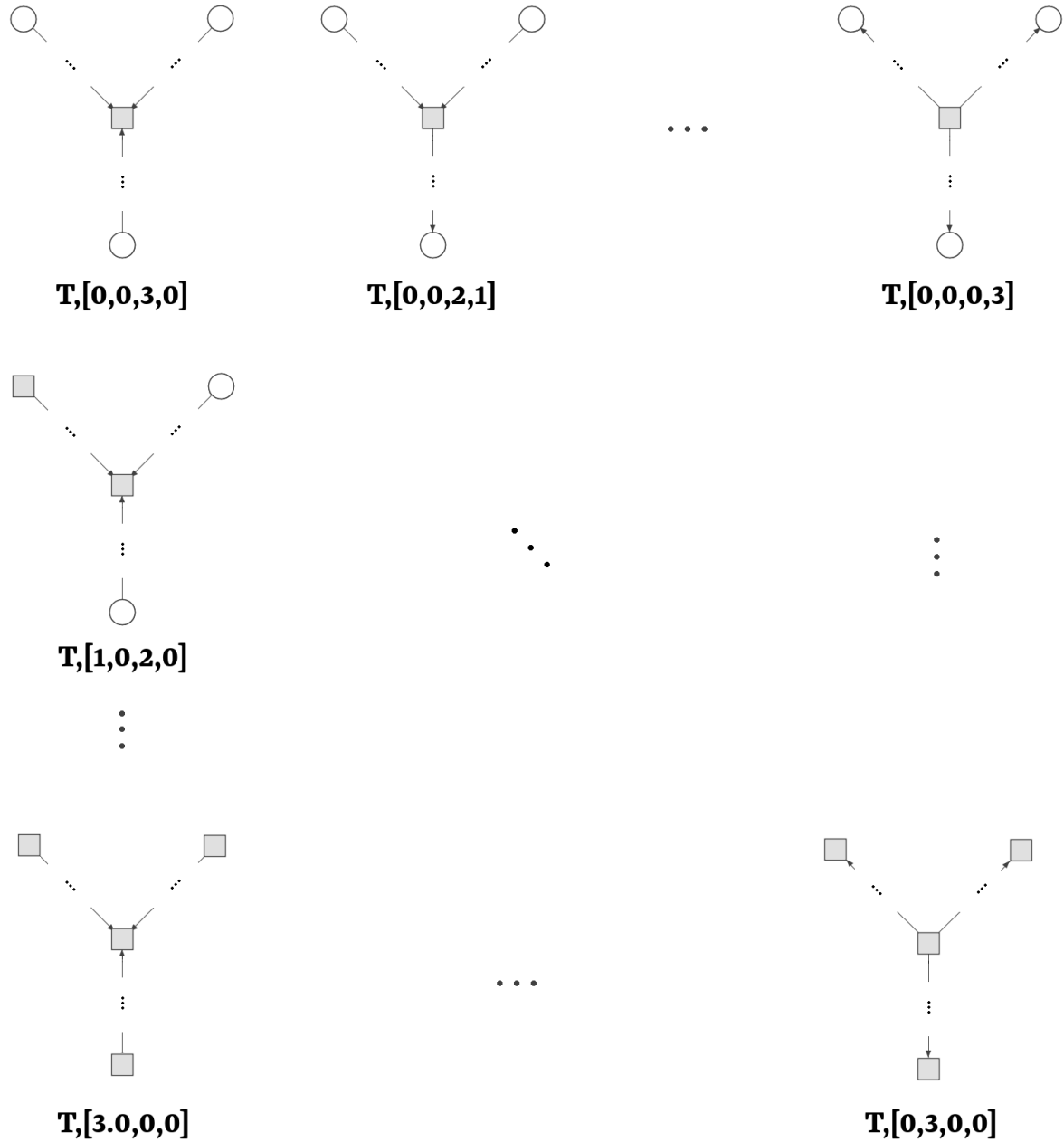
8.1 Wprowadzenie

W niniejszym rozdziale przedstawiono porównywanie sieci Petriego za pomocą wierzchołków rozgałęziających, które zostały zdefiniowane w Rozdziale 5. Do porównania wykorzystano Dystans Względnej Częstotliwości, który został dostosowany do operowania na parach uporządkowanych L (wprowadzonych w równaniu 5.8) zawierający na pierwszej pozycji typ wierzchołka rozgałęziającego przedstawiony za pomocą litery T lub P, a na drugiej pozycji zawierający wektor czterech nieujemnych liczb odpowiadających liczbie końcówek danego wierzchołka rozgałęziającego.

8.2 Pomysł i inspiracja

Wierzchołki rozgałęziające zostały zdefiniowane w Rozdziale 5. Pierwszy kontakt z ich konceptem pojawił się podczas prac nad metodą z Rozdziału 9, gdzie stanowiły rdzeń proponowanego algorytmu. To właśnie świadomość ich istotności dla niezmienników i struktur bazujących na nich spowodowała, że podczas późniejszych prac nad implementacją metod grafletowych zaczęły być rozważane jako podstawa do tworzenia struktur dedykowanych dla sieci Petriego.

Wierzchołek rozgałęziający dowolnego typu samodzielnie przedstawia niewystarczający zbiór informacji o strukturze modelu. Nawet dla zredukowanych sieci sprawdzenie wierzchołków rozgałęziających odpowiada tylko za warunek konieczny dla istnienia izomorfizmu. Oczywiście więc była potrzeba wyjścia poza pojedynczy wierzchołek i opisanie sąsiedztwa oraz jego funkcji w sieci. Na podstawie analiz przeprowadzonych na podsieciach bazujących na t -niezmiennikach (Rozdział 4) została stwierdzona relacja pomiędzy dwoma wierzchołkami rozgałęziającymi połączonymi bezpośrednią ścieżką (jest to ścieżka zawierająca wierzchołki rozgałęziające jedynie na pierwszej i ostatniej pozycji). Bazując na tym został opisany zbiór końcówek dla każdego wierzchołka rozgałęziającego (wzór (5.6)). Konieczne jest też rozróżnienie typu wierzchołka rozgałęziającego ze względu na jego różną rolę i wpływ na działanie sieci. Pod wpływem analiz przeprowadzonych dla dekompozycji i redukowanych sieci, celowo pomijane zostają wierzchołki przepływowe występujące pomiędzy wierzchołkami rozgałęziającymi a ich końcówkami. Pozwala to na pominięcie różnic niewpływających na liczbę i formę niezmienników w porównywanych sieciach.



Rysunek 8.1: Przykład podsieci o rozmiarze cztery reprezentowanych przez wierzchołki rozgałęziające i ich końcówki.

Opracowaną metodę zliczającą podsieci oparte o wierzchołki rozgałęziające wraz z ich końcówkami można porównać do metod określających podobieństwo przez zliczanie drzew w grafach. Należy jednocześnie wziąć pod uwagę, że choć struktury indukowane przez wspomniane wierzchołki często będą miały formę drzewa, nie jest to jedyny rodzaj struktur jaki może zostać utworzony na ich bazie.

8.3 Porównanie z grafletami

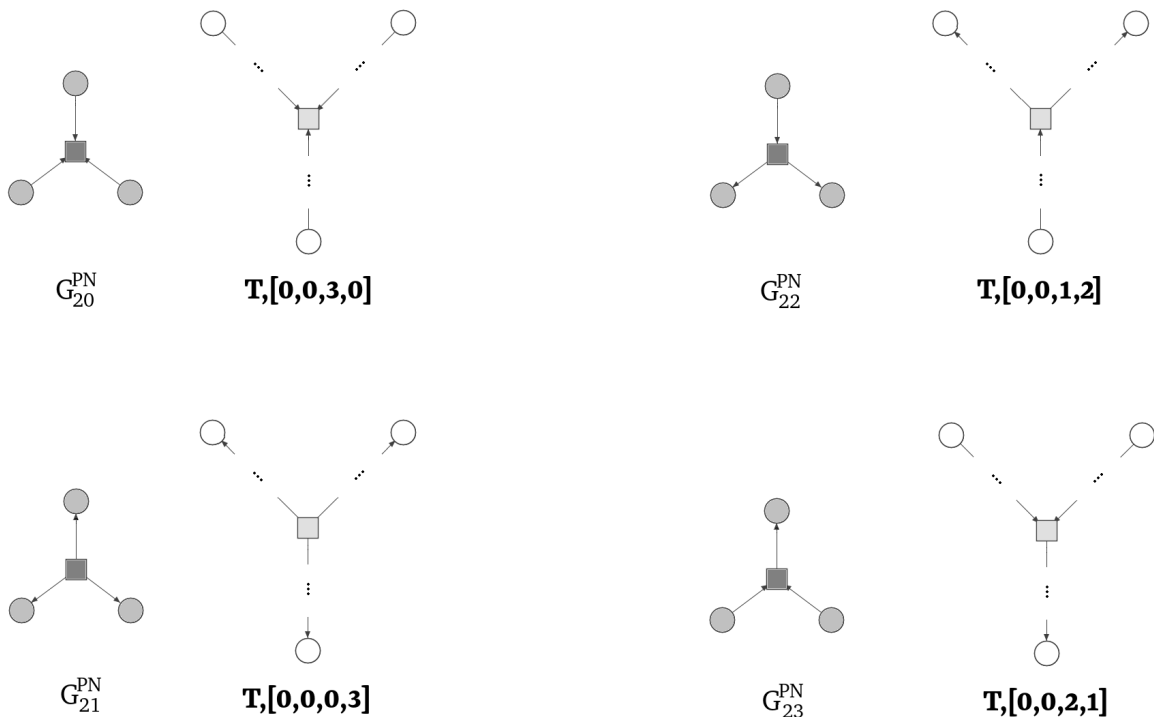
W przypadku wierzchołka rozgałęziającego i jego końcówek przez indukowaną podsieć rozumiemy podsieć zawierającą wspomniane wierzchołki oraz wszystkie miejsca i tranzycje istniejące w ścieżkach łączących korzeń z końcówkami. Jednak informacja na temat tranzycji i miejsc przepływowych jako

elementów podlegających redukcji jest z perspektywy proponowanej metody pomijalna. Natomiast kluczowe dla porównania informacje zawarte są w zbiorze wierzchołków rozgałęziających wraz z kierunkiem połączenia z korzeniem.

Na Rysunku 8.1 przedstawiono zapis podsieci indukowanych czterema wierzchołkami rozgałęziającymi w uproszczonej postaci graficznej. Dla każdego przypadku zawiera ona wierzchołek rozgałęziający pełniący rolę korzenia, oraz jego końcówki połączone przerywanymi łukami. Przy takiej reprezentacji graficznej może dojść do sytuacji, że dwa wierzchołki tego samego typu są połączone przerywanym łukiem, np. $T, [1, 0, 2, 0]$. Taka sytuacja nie jest sprzeczna z teorią sieci Petriego, ponieważ przerywany łuk reprezentuje ścieżkę łączącą dwa wierzchołki rozgałęziające a nie pojedynczy łuk.

W przypadku występowania pomiędzy wierzchołkami różnych typów może opisywać ona ścieżkę długości odpowiadającej liczbie nieparzystej, a liczbie parzystej w przypadku występowania pomiędzy wierzchołkami tego samego typu. Z tego powodu za pomocą takiej struktury można reprezentować wiele podsieci redukowalnych do wspólnej postaci – przykład pokazany jest na Rysunku 5.2.

Podsieci indukowane przez wierzchołki rozgałęziające i pn-graflety reprezentują małe struktury zbudowane na kilku wierzchołkach. Z artykułu [120] wiemy o 151 pn-grafletach stosowanych do porównywania grafów. Odpowiadać im będzie 118 podsieci indukowanych przez wierzchołki rozgałęziające o rozmiarach od 2 do 5 wierzchołków (230 w przypadku rozszerzenia do rozmiaru 6). Dokładny przyrost liczby podsieci został przedstawiony w Tabeli 8.1. Na jej podstawie można zauważyć, że wraz ze wzrostem rozmiaru pn-grafletu i struktury indukowanej przez wierzchołki rozgałęziające, pn-graflety mają znacznie większy przyrost możliwych struktur. Zawdzięczają to możliwości reprezentacji połączeń pomiędzy wszystkimi wierzchołkami, a nie jak w przypadku sieci indukowanych przez wierzchołki rozgałęziające, tylko pomiędzy korzeniem a końcówkami.



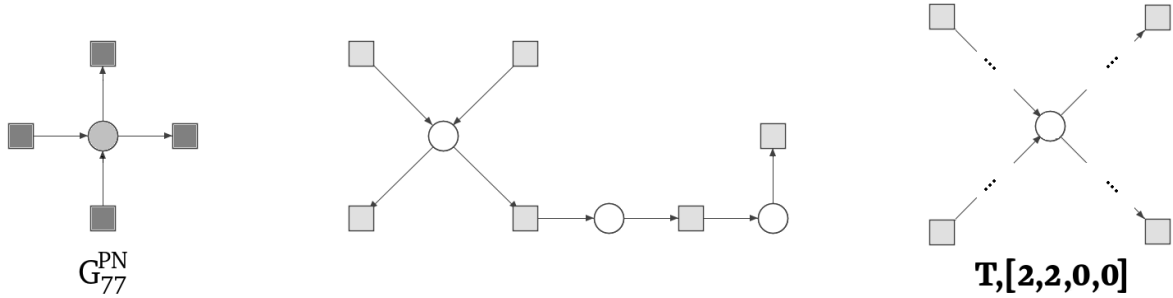
Rysunek 8.2: Cztery graflety o rozmiarze 4 reprezentujące drzewa z tranzycją jako korzenie, oraz odpowiadające im podsieci indukowane przez tranzycje rozgałęziające.

Podsieci indukowane przez wierzchołki rozgałęziające, wyznaczone dla sieci na których przeprowadzono redukcję ścieżek (redukcja strukturalna numer 7) i pn-graflety odpowiadają podsieciom o małych rozmiarach. W szczególnych przypadkach będą one odpowiadać tym samym podsieciom. Przykład takiej sytuacji przedstawiono na Rysunku 8.2, gdzie występują cztery pn-graflety $G_{20}^{PN} - G_{23}^{PN}$ i odpowiadające im na zredukowanych sieciach cztery struktury oparte na tranzycjach rozgałęziających.

Tabela 8.1: Porównanie ilościowe pn-grafletów i podsieci indukowanych przez wierzchołki rozgałęziające w zależności od ich rozmiaru (w przypadków wierzchołków rozgałęziających przez rozmiar określany jest przez liczbę końcówek $|B(v^{br})| + 1$).

Liczba Wierzchołków	pn-graflety	Wierzchołki Rozgałęziające
2	2	–
3	6	8
4	23	40
5	120	70
6	?	112
suma Σ	151	118 (230 rozmiaru 6)

Natomiast w przypadku porównywania niezredukowanych sieci za pomocą pn-grafletów i podsieci indukowanych wierzchołkiem rozgałęziającym oba podejścia będą nadal wykrywać te same struktury. Różnicą będą natomiast dodatkowe wystąpienia pn-grafletów reprezentujące niezredukowane obszary sieci. Jako przykład takiej sytuacji posłużą trzy podsieci z Rysunku 8.3. Są to pn-graflet G_{77}^{PN} , podsieć indukowana miejscem rozgałęziającym $P, [2, 2, 0, 0]$, oraz uproszczona graficzna reprezentacja $P, [2, 2, 0, 0]$. Po redukcji polegającej na skróceniu ścieżek do minimalnej długości (redukcja numer 7 z Rozdziału 6), środkowa podsieć odpowiadająca $P, [2, 2, 0, 0]$ będzie izomorficzna z pn-grafletem G_{77}^{PN} . Natomiast jeśli redukcja nie zostanie przeprowadzona pn-graflet G_{77}^{PN} nadal będzie występować w tej podsieci, razem z innymi pn-grafletami nie odpowiadającymi swoją strukturą $P, [2, 2, 0, 0]$ (np. pn-graflet G_{55}^{PN}).



Rysunek 8.3: Trzy struktury odpowiadają kolejno pn-grafletowi G_{77}^{PN} , podsieci zbudowanej wokół miejsca rozgałęziającego, oraz graficznej reprezentacji pary uporządkowanej $P, [2, 2, 0, 0]$.

Dla grafletów $G_{16}^{PN} - G_{19}^{PN}$ zachowana jest taka sama relacja tylko dla miejsc rozgałęziających będących korzeniem. Jest to jednak stosunkowo mały zbiór przypadków, gdzie za pomocą dwóch różnych podejść opisywana jest taka sama podsieć. Dla pozostałych 14 grafletów o rozmiarze 4 sytuacja jest bardziej skomplikowana. Graflety $G_8^{PN} - G_{15}^{PN}$ i G_{25}^{PN} odpowiadają ścieżkom (lub ich częściom) występującym pomiędzy dwoma dowolnymi wierzchołkami rozgałęziającymi. Ponadto podlegają one redukcjom przedstawionym w punktach 7 oraz 8 na Rysunku 2.2 i z tego powodu ich wystąpienia nie są wykrywane za pomocą wierzchołków rozgałęziających. Natomiast pozostałe odpowiadają przypadkom sieci indukowanych przez wierzchołki rozgałęziające rozmiaru 2 lub kombinacji dwóch takich podsieci.

Wniosek z powyższej analizy jest następujący – pn-graflety i sieci indukowane przez wierzchołki

rozgałęziające, mimo że bazują na małych podsieciach i wykorzystują tę samą metrykę (modyfikacje RGF i RBF) do wyznaczania podobieństwa, badają różne własności strukturalne i nie są zamienne w zastosowaniu.

8.4 Algorytm

Wstępne metody porównywania

W ramach badań naukowych prowadzonych na potrzeby niniejszej rozprawy opracowane zostały cztery heurystyki do porównywania sieci, które wykorzystują wierzchołki rozgałęziające do określenia skali podobieństwa pomiędzy sieciami. W swej formie przypominają one warianty algorytmu opisanego w Rozdziale 7 porównującego niezmienniki poprzez dopasowanie ich w pary. Różnice pomiędzy wstępnymi wariantami heurystyk opierały się na dwóch płaszczyznach. Pierwszą był stopień rozróżnienia końcówek na podstawie typu wierzchołka, lub stopnia wierzchołka, lub postaci zapisanej wzorami 5.7 i 5.8. Drugą była kwestia zawierania się w sobie podsieci indukowanych przez wierzchołki rozgałęziające.

Uzyskane rezultaty okazały się niesatysfakcjonujące. Testy wykonano na sieciach znanych z eksperymentów przeprowadzonych dla GDDA w Rozdziale 6. Niestety, żaden z wariantów nie wykazał odpowiedniej skuteczności, odbiegając jakością wyników od metryk grafletowych GDDA i RGF. W efekcie, po ponownym przeszukaniu dostępnej literatury, uwagę zwróciły metryki RBF i RGF. Podczas testów pn-grafletów znalazły się w cieniu GDDA oraz testów przeprowadzanych z ich użyciem, natomiast ich postać okazała się idealna do zastosowania dla wierzchołków rozgałęziających.

RBF – adaptacja metody grafletowej

Modyfikacja metody RBF (ang. *Relative Branching Vertices Frequency distance*), podobnie jak w przypadku adaptacji RGF przeznaczonej dla pn-grafletów, polegała na zmianie zbioru wejściowego. Różnicą w porównaniu do RGF jest liczba zliczanych wierzchołków (struktur). W przypadku RGF poszukujemy ustalonego wcześniej zbioru pn-grafletów o stałym rozmiarze (dla rozmiaru do 5 wierzchołków mowa o 151 pn-grafletach). Natomiast w przypadku RBF, ze względu na specyfikę wierzchołków rozgałęziających i podsieci, jakie indukują wraz ze swoimi końcówkami, zdecydowano się na dynamiczne wyznaczanie zbioru podsieci używanych do porównywanych sieci. W wyniku tego zbiór podsieci, których rozkład bada RBF, nie będzie miał stałej liczby elementów, jak to ma miejsce w przypadku 151 pn-grafletów rozmiaru 5 używanych przez RGF. Zamiast tego, liczba typów podsieci analizowanych przez RBF będzie równa liczbie unikalnych wierzchołków rozgałęziających zapisanych w postaci $L(v^{br}) = (f(v^{br}), [|T_{in}^{v^{br}}|, |T_{out}^{v^{br}}|, |P_{in}^{v^{br}}|, |P_{out}^{v^{br}}|])$ z Równań 5.8 i 5.7, które są obecne w obu porównywanych sieciach. Oznacza to, że zbiór ten będzie dynamicznie zmieniał się w zależności od specyfiki porównywanych sieci, co stanowi główną różnicę między RGF a RBF i wpływa na rodzaj danych wejściowych używanych w tych metodach. To jest główna różnica pomiędzy RGF a RBF polegająca na zmianie rodzaju danych wejściowych.

Do wyznaczenia wartości RBF wykonujemy następujące kroki, gdzie b reprezentuje liczbę rodzajów różnych podsieci indukowanych przez wierzchołki rozgałęziające i ich końcówki:

$$T(G) = \sum_{i=1}^b N_i(G) \quad (8.1)$$

gdzie $N_i(G)$ zwraca liczbę wystąpień w sieci G wierzchołka rozgałęziającego zapisanego za pomocą pary uporządkowanej L z równania 5.8.

$$F_i(G) = -\log(N_i(G)/T(G)) \quad (8.2)$$

gdzie $F_i(G)$ jest normalizacją wartości $N_i(G)$.

$$D(G, H) = \sum_{i=1}^b |F_i(G) - F_i(H)| \quad (8.3)$$

gdzie $D(G, H)$ określa sumę znormalizowaną dystansów pomiędzy liczbą wystąpień każdej pary uporządkowanej L w dwóch porównywanych sieciach G i H .

Używając nowych struktur opartych na wierzchołkach rozgałęziających nie jest koniecznym ograniczenie się do n struktur, np. wszystkich o rozmiarze do 6 wierzchołków, jak to ma miejsce przy pn-grafletach. Jest to spowodowane tym, że proces znalezienia wszystkich grafletów w sieci jest znacznie bardziej złożony obliczeniowo w porównaniu z wyznaczeniem wszystkich wierzchołków rozgałęziających wraz z końcówkami. Jednocześnie można wyznaczyć wszystkie rodzaje struktur rozgałęziających istniejących w sieciach, co nie jest możliwe dla grafletów.

8.5 Przykłady

8.5.1 Małe sieci metaboliczne

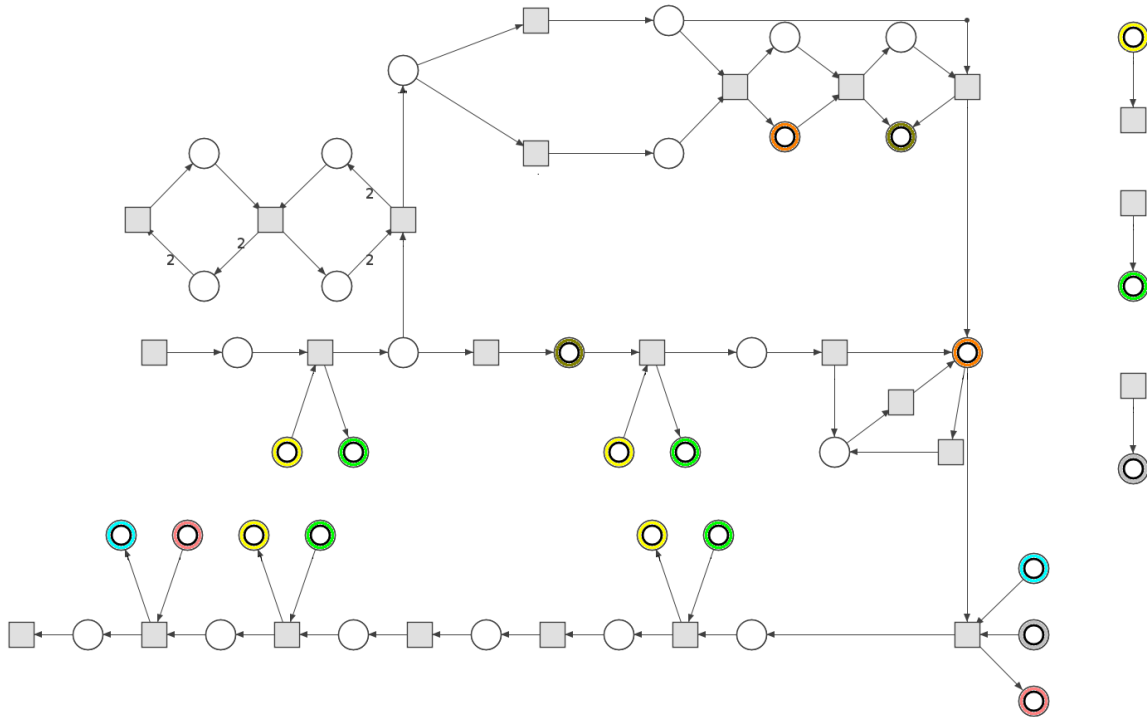
Do przedstawienia zachowania metryki RBF na małych sieciach (poniżej 50 wierzchołków) wybrano trzy modele różnych szlaków metabolicznych z artykułu [51], przedstawionych na Rysunkach 8.4-8.6. Wyniki trzech porównań z podziałem na wystąpienia każdego typu wierzchołka rozgałęziającego zostały zwizualizowane na diagramach z Rysunku 8.7. Wartości są przedstawione z perspektywy wierzchołków, tranzycji i miejsc. Dodatkowo załączony jest wykres z wynikami RGF dla każdego pn-grafletu występującego w tych sieciach jako punkt odniesienia.

Jako że metoda operuje na dystansie pomiędzy występowaniem konkretnych wierzchołków rozgałęziających, analizując wartości RBF konieczne jest wzięcie pod uwagę nie tyle ogólnego rozmiaru porównywanych sieci, ile sumarycznej liczby wystąpień rozgałęziających wierzchołków.

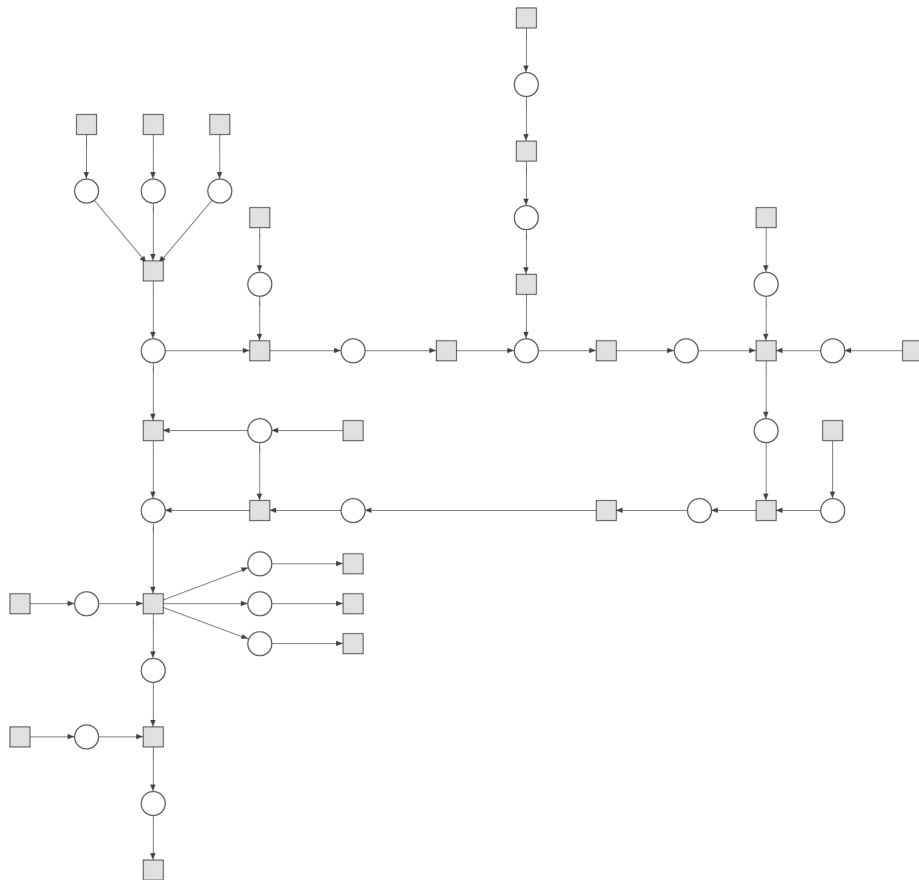
Tabela 8.2: Porównanie wyników RBF z podziałem na wartości dla wierzchołków, tranzycji i miejsc rozgałęziających dla wszystkich par sieci SM I, SM II, SM III z pracy [51]. Dla porównania zamieszczono wyniki RGF dla tych samych przypadków.

Porównania	RBF			RGF
	Wierzchołki	Tranzycje	Miejsca	Graflety
SM I - SM II	60,61	25,77	19,40	328,93
SM I - SM III	63,21	16,31	29,58	190,00
SM II - SM III	53,76	20,10	19,87	268,01

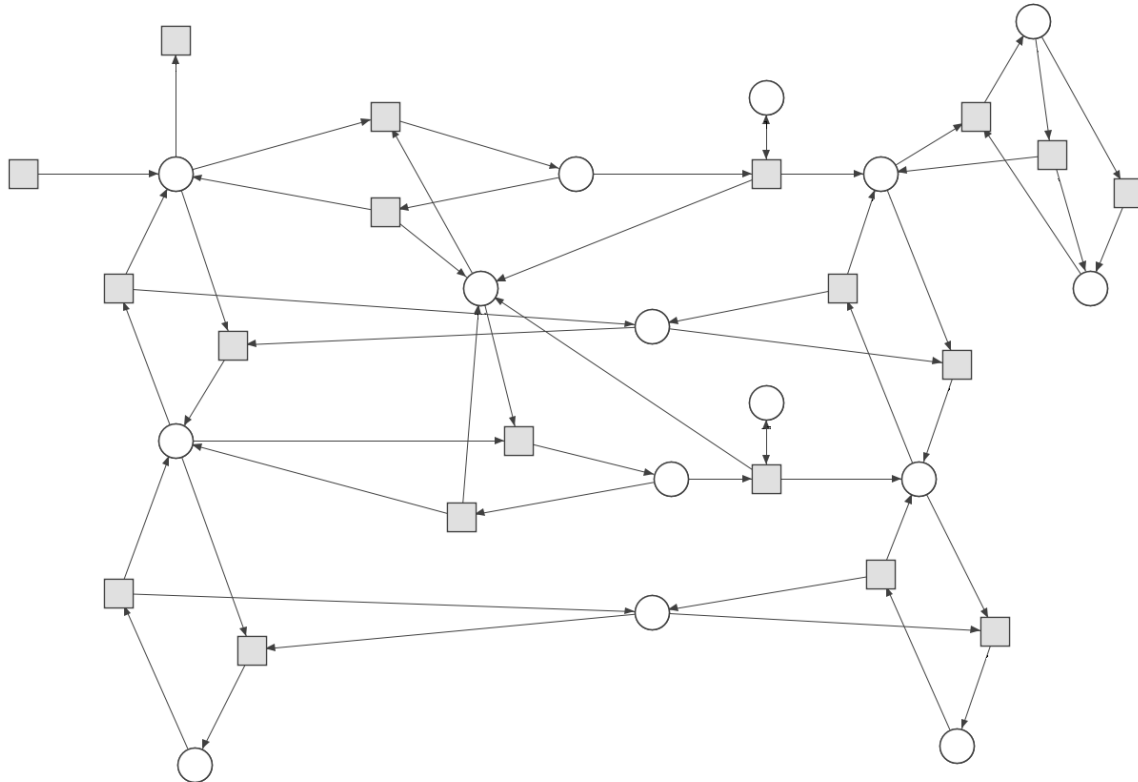
Testowane przykłady reprezentują sieci podobnego rozmiaru i liczby wierzchołków rozgałęziających V^{br} – szczegółowe wyniki porównania przedstawiono w Tabeli 8.2. Wartość ogólna RBF dla wszystkich typów wierzchołków (kolumna Wierzchołki) dla porównań sieci SM I z SM II i SM I z SM III



Rysunek 8.4: Prosta sieć metaboliczna SM I z artykułu [51]



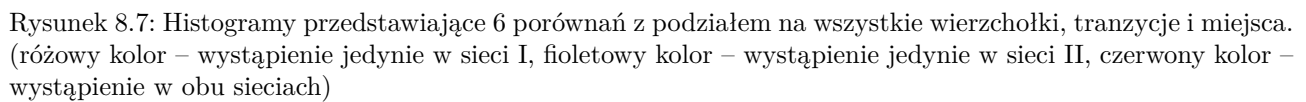
Rysunek 8.5: Prosta sieć metaboliczna SM II z artykułu [51]



Rysunek 8.6: Prosta sieć metaboliczna SM III z artykułu [51]

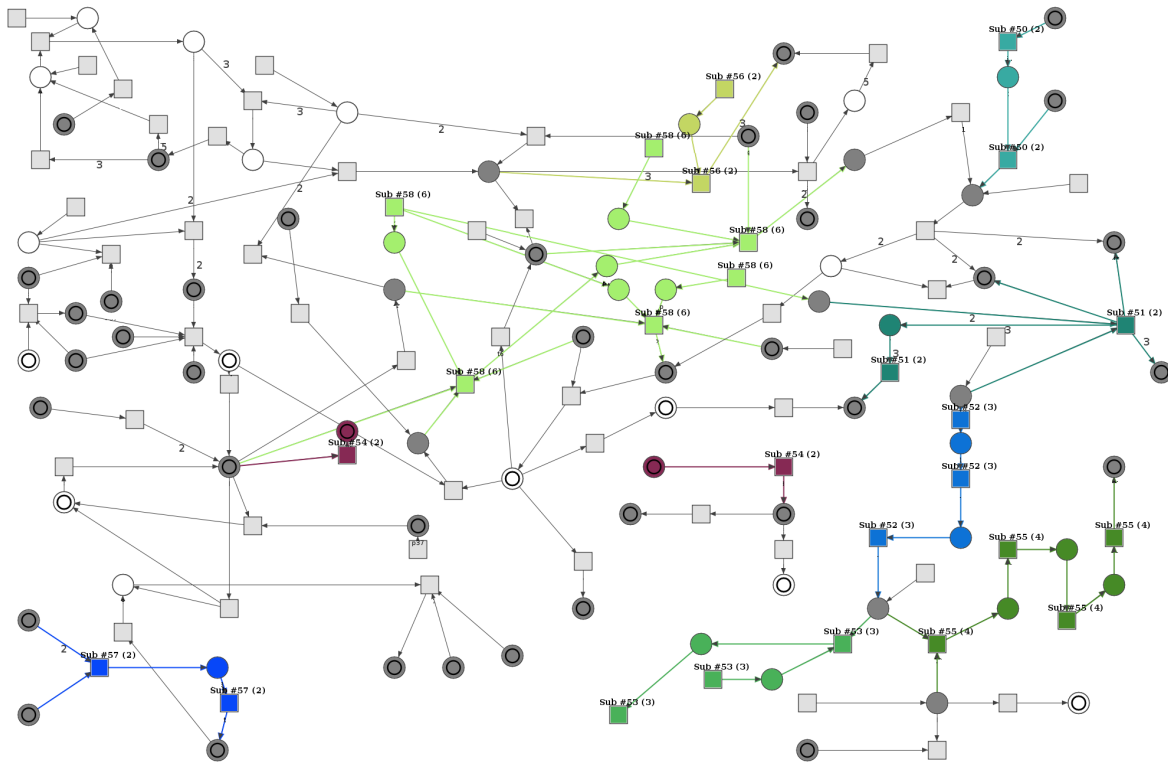
są zbliżona. Porównanie sieci SM II z SM III wyróżnia się mniejszym dystansem od dwóch pozostałych, wciąż jednak reprezentuje ono zbliżony poziom różnic. W przypadku podziałów na konkretne typy wierzchołków rozgałęziających (kolumny Tranzycje i Miejsca) widoczne są różne dystanse, za które odpowiadają różne stosunki liczby tranzycji do liczby miejsc rozgałęziających. W modelach systemów biologicznych przedstawionych z użyciem sieci Petriego w zdecydowanej większości przypadków występuje zbliżona liczba miejsc i tranzycji. Jest możliwa dominacja jednego typu, jednak w modelach opisywanych w literaturze występuje tylko niewielka nadreprezentacja tranzycji względem miejsc. Natomiast w przypadku wierzchołków rozgałęziających dwudzielność nie ma już takiego wpływu na równomierne występowanie tranzycji i miejsc rozgałęziających, w wyniku czego możliwe do zaobserwowania będą porównania sieci, w których jeden typ wielokrotnie przeważa nad drugim (lub jeden typ wierzchołków rozgałęziających nie istnieje w sieci). Analizując dane cząstkowe RBF przedstawione na wykresach z Rysunku 8.7 wyraźnie widać niski stopień podobieństwa pomiędzy parą porównywanych sieci (czerwone fragmenty histogramu odpowiadające elementom występującym w obu modelach). Jedynie pojedyncze wierzchołki rozgałęziające tego samego rodzaju występują jednocześnie w porównywanych parach. Dodatkowo w przypadku tranzycji rozgałęziających dla porównania SM I z SM II reprezentują one rozłączne zbiory.

Porównując wyniki uzyskane dla metod RGF korzystającego z pn-grafletów i RBF korzystającego z wierzchołków rozgałęziających, można zaobserwować, że mimo podobieństwa metod, wyniki przedstawiają dystanse na dwóch różnych rzędach wielkości. Jest to spowodowane rozmiarem podsieci, które są wykorzystywane do porównania. Jednocześnie w obu przypadkach widoczne są znaczące różnice

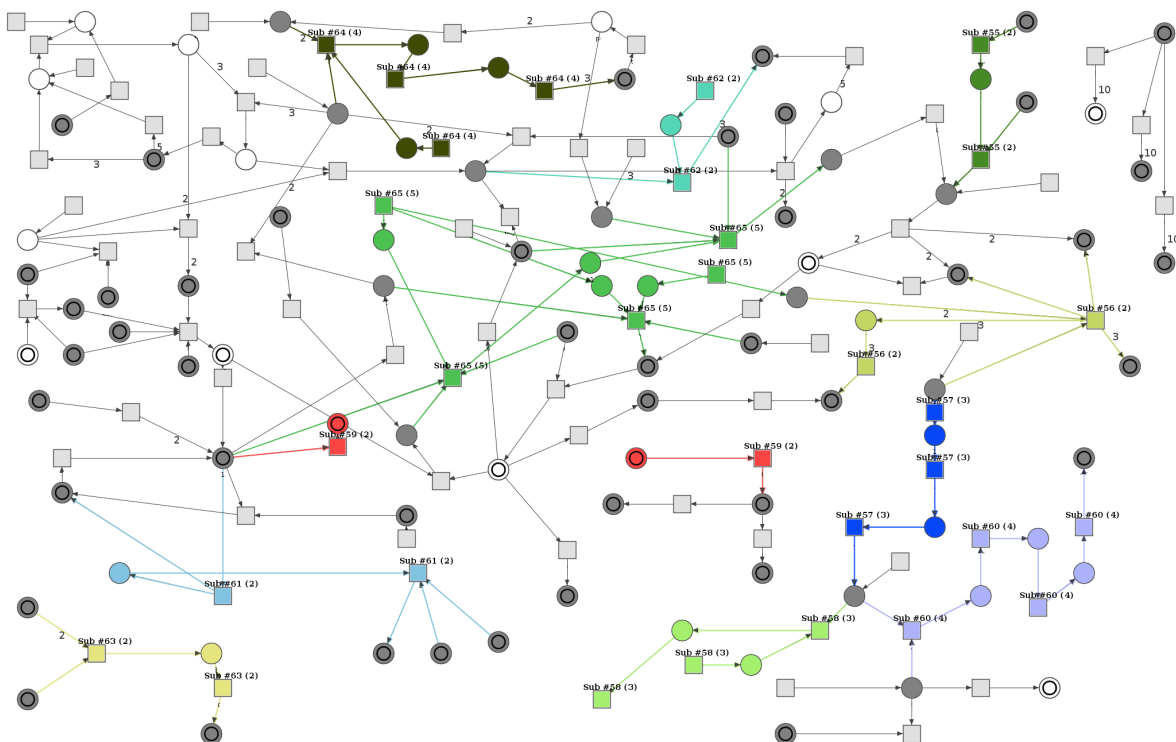


Porównując wyniki uzyskane dla RBF z tymi uzyskanymi dla RGF, za atut RBF można uznać mniejszą liczbę struktur, których rozkład w sieci jest poszukiwany oraz większe niż w przypadku pojedynczego pn-grafletu znaczenie wystąpienia wierzchołka rozgałęziającego danego typu. Mniejsza liczba wystąpień wierzchołków rozgałęziających, która w pesymistycznym przypadku będzie równa liczbie wierzchołków w sieci, daje możliwość szczegółowej analizy poszczególnych wystąpień, co w przypadku pn-grafletów i ich rozkład często zawierającej kilkadziesiąt tysięcy pn-grafletów dla sieci o rozmiarze 50-80 wierzchołków jest niepraktyczne.

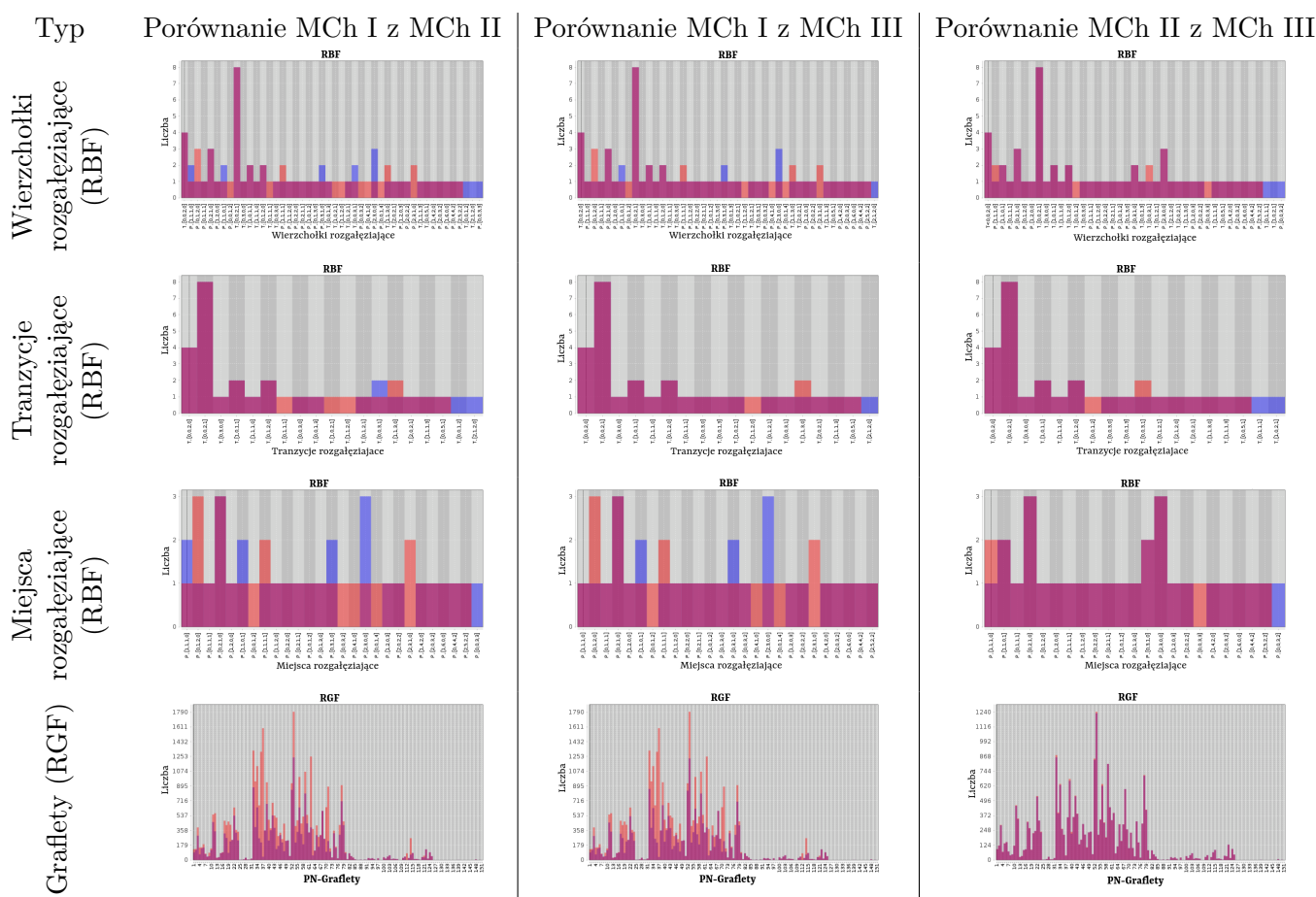
W tym podrozdziale zostanie rozważony przypadek odwrotny do poprzedniego, reprezentujący trzy sieci: MCh I (Rysunek 8.8), MCh II (Rysunek 8.9) i MCh III (Rysunek 7.2) reprezentujące różne warianty modelu pacjenta chorego na nadciśnienie i, w przypadku MCh II i MCh III, z infekcją SARS-CoV-2. W efekcie znany jest rozmiar największego wspólnego podgrafu pomiędzy każdą parą modeli, który może być użyty jako punkt odniesienia dla skali podobieństwa struktur (w wyniku wyłączenia usunięty zostaje fragment sieci, co można interpretować jako powstanie drugiej sieci, która zawiera się w pierwszej i jest tym samym ich największym wspólnym podgrafem).



Rysunek 8.8: Model przedstawiający pacjenta chorego na nadciśnienie, ale bez infekcji SARS-CoV-2, która została wyłączona. Na sieci zaznaczono nietrywialne t-sieci. Sieć oznaczono skrótem MCh I



Rysunek 8.9: Model przedstawiający pacjenta chorego na nadciśnienie i z infekcją SARS-CoV-2. Na sieci zaznaczono nietrywialne t-sieci. Sieć oznaczono skrótem MCh II



Rysunek 8.10: Seria porównań trzech dużych sieci reprezentujących organizm pacjenta chorego na nadciśnienie i w przypadku MCh II, MCh III, z infekcją SARS-CoV-2.

Analizując rezultaty z Tabeli 8.3 i Rysunku 8.10 łatwo wskazać, że dla porównania sieci MCh II i MCh III obserwujemy niski dystans dla wszystkich przypadków, a zwłaszcza grafletów. Wartość RBF odnosi się tylko do różnic strukturalnych powstałych bezpośrednio przez operację wyłączenia: wierzchołków usuniętych z modelu oraz wierzchołków połączonych z usuniętymi, dla których nastąpiła zmiana stopnia.

Tabela 8.3: Porównanie wyników RBF z podziałem na osobne wartości dla wierzchołków, tranzycji i miejsc rozgałęziających dla porównań sieci MCh I, MCh II, MCh III. Dla porównania zamieszczono wyniki RGF dla tych samych przypadków.

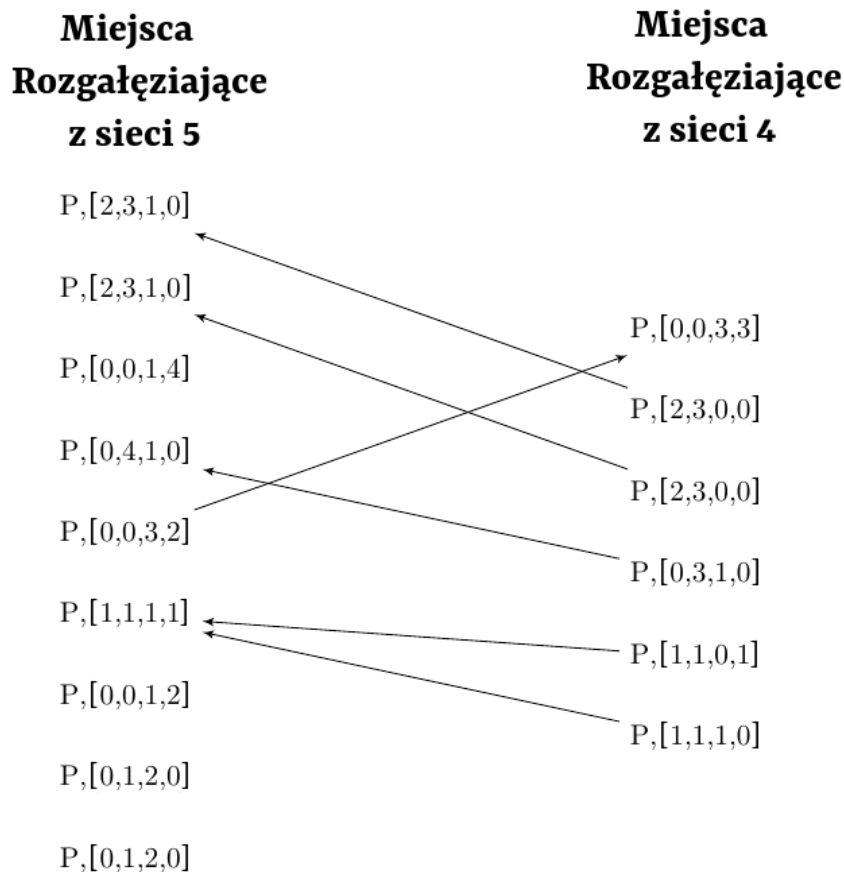
Porównania	RBF			RGF
	Wierzchołki	Tranzycje	Miejsca	Grafety
MCh I-MCh II	52,79	18,73	26,64	60,34
MCh I-MCh III	31,64	7,93	19,85	61,037
MCh II-MCh III	22,19	10,79	8,05	1,02

Aby sklasyfikować, które wierzchołków rozgałęziających występujących tylko w jednej sieci powstały jako efekt operacji wyłączenia podsieci, użyto grafu zawierania.

Graf zawierania

Graf zawierania reprezentuje sytuację w której podsieć indukowana przez wierzchołek rozgałęziający z sieci A zawiera się w sieci indukowanej przez wierzchołek rozgałęziający z sieci B (dla każdej pozycji w wektorze $E(v_i)$ wartość jest mniejsza lub równa wartości z odpowiadającej pozycji w wektorze $E(v_j)$). Jest to w grafie zawierania reprezentowane przez łuk łączący v_i z wierzchołkiem v_j . Główną funkcją grafu zawierania jest detekcja potencjalnych redukcji oraz wierzchołków stycznych do różnic strukturalnych, np. reprezentujących wyłączenie fragmentu sieci.

Na Rysunku 8.11 przedstawiono miejsca rozgałęziające z porównania MCh I-MCh II, które występują tylko w jednej sieci lub liczba ich wystąpień jest mniejsza niż w drugiej. Łukami zaznaczono zawieranie się miejsc przy ograniczeniu do jednego stopnia wejściowego lub wyjściowego różnicy pomiędzy wierzchołkami (jako przykład miejsce rozgałęziające $P, [2, 3, 1, 0]$ zawierające jako końcówkę jedną tranzycję wejściową, w przeciwieństwie do $P, [2, 3, 0, 0]$ która jej nie posiada, ale ze względu że pozostałe wartości są identyczne, podsieć indukowana przez nią może zawierać się w podsieci indukowanej przez $P, [2, 3, 1, 0]$). W zależności od potrzeb można zwiększyć ograniczenie, aby wychwycić miejsca będące punktem stykowym kilku (wyłączonych tym przypadkiem) podsieci.



Rysunek 8.11: Przykład grafu zawierania dla miejsc rozgałęziających występujących tylko w sieci 4 lub tylko w sieci 5 (porównanie miejsc sieci 4 i 5 z Rysunku 8.10)

W rezultacie możemy wskazać 5 zawierania wierzchołków z modelu MCh I w MCh II i jednego wierzchołka z MCh II w MCh I, które mają potencjalne połączenie z podsiecią lub łukiem, nieobecnym w jednej z porównywanych sieci.

Samo zawieranie się wierzchołków rozgałęziających nie oznacza podobieństwa biologicznego pomiędzy nimi, a strukturalne. Wsparcie się dystansem grafowym w celu określenia, które z przypadków są bliższe izomorfizmowi struktur także nie jest idealnym podejściem. Sugerowanym jest, aby kwestię podobieństwa pomiędzy wierzchołkiem zawierającym i zawierającym rozwiązać z użyciem transformacji znanych z Grafowego Dystansu Edycyjnego. Konkretnie transformacje, jak np. rozdwojenie końcówki, czy dodanie końcówki nowego typu, mogą zostać ocenione jako bardziej prawdopodobne w systemie biologicznym, a w wyniku tego o tańszym koszcie transformacji. Do tego dochodzą przypadki znane z redukcji sieci Petriego, które mogą dostać osobną ocenę kosztu w zależności od porównywanych sieci. Dzięki takim operacjom możliwe będzie dokładniejsze wyselekcjonowanie par wierzchołków, które przedstawiają podobne podprocesy z perspektywy systemu biologicznego.

8.6 Podsumowanie

RBF bazujący na strukturach opartych o wierzchołki rozgałęziające jest alternatywą dla RGF używającego pn-grafletów. Jest to widoczne zwłaszcza w przypadku porównania dużych sieci, dla których konieczność znalezienia wszystkich grafletów jest trudna obliczeniowo. Zamiast uciekać się do użycia heurystyk w celu oszacowania liczby grafletów w sieciach można skorzystać z RBF, dla którego pesymistyczna złożoność czasowa wynosi $O(n^2)$, gdzie n jest liczbą wierzchołków.

Jednocześnie należy zaznaczyć, że mimo, iż oba podejścia różnią się *tylko* zbiorem zliczanych podsieci, ma to znaczący wpływ na ich możliwości detekcyjne różnic w porównywanych sieciach oraz interpretację wyniku. Liczba znalezionych wierzchołków rozgałęziających będzie zazwyczaj znacząco mniejsza od liczby wszystkich wierzchołków w sieci (w szczególnych przypadkach może być jej równą). Natomiast w przypadku pn-grafletów sytuacja jest odwrotna, tj. liczba znalezionych w sieci pn-grafletów rzadko będzie oscylować w okolicy jej rozmiaru. Zazwyczaj będzie przewyższać ją od kilkudziesięciu do kilkuset razy. Z tych właśnie powodów wartości RBF i RGF muszą być różnie interpretowane.

Metoda RBF (oraz RGF) może być wykorzystana do wstępnego określenia podobieństwa strukturalnego pomiędzy elementami ze zbioru sieci. Pozwala ona na eliminację przypadków nieprzejawiających znacznego podobieństwa struktur, dostarczając zbiór, na którym można przeprowadzić bardziej dokładne i co za tym idzie złożone analizy. Drugim przypadkiem zastosowania jest porównanie dużych modeli opartych na sieciach Petriego, dla których występuje problem wykładniczego wzrostu liczby cząstkowych wektorów, który zapobiega wykorzystaniu innych dostępnych metod opartych o niezmienniki z Rozdziałów 7 i 9.

Ze względu na ograniczenie się tylko do wierzchołków rozgałęziających RBF jest odporna na różnice w postaci długości ścieżek występujących pomiędzy wierzchołkami rozgałęziającymi (przyczyną różnicy jest różny poziom szczegółowości wybrany przez autorów porównywanych modeli). Dodatkowo z użyciem grafu zawierania możliwe jest wykrycie przypadków podlegających redukcjom opisanym w podrozdziale 2.2.3.

Rozdział 9

Porównanie przez dekompozycję – podsieci t-sieć, indukowane przez conADT, MCT i podsieci funkcyjne

9.1 Wprowadzenie

W niniejszym rozdziale zostaje opisana rodzina metod porównywania sieci Petriego opartych o dwa mechanizmy, tj. dekompozycję do rozłącznych podsieci i poszukiwanie największej wspólnej podsieci będącej potencjalnie interesującą z biologicznej perspektywy.

9.1.1 Krótka historia rozwoju

Pomysł, który leży u podwalin metod prezentowanych w tym rozdziale i następnym, powstał na samym początku badań nad porównywaniem sieci Petriego opisywanych w tej rozprawie. Pierwsza iteracja algorytmu porównywania bazowała na podziale obu porównywanych modeli na zbiory rozłącznych podsieci *conADT*, gdzie porównanie następowało pomiędzy parą podsieci, dla których znajdowana była największa wspólna podsieć oparta o wierzchołki rozgałęziające. Dokładna definicja wierzchołków rozgałęziających została opracowana dopiero kilka lat po implementacji pierwszego prototypu. Algorytm w swej wstępnej formie ukazał się w pracy [123]. Następnie prace nad rozwojem i dopracowaniem metody zostały zamrożone na okres trzech lat. Jest to o tyle ważne, że w danym okresie rozpoczęte zostały badania nad przeglądem istniejących metod dekompozycji sieci Petriego oraz opracowano definicję wierzchołków rozgałęziających na potrzeby metody opisanej w Rozdziale 8. Efektem był powrót do koncepcji algorytmu porównującego zdekomponowane podsieci i w oparciu o uzyskaną wiedzę rozszerzenie do działania na pokrewnych dekompozycjach oraz prawidłowe zdefiniowanie poszukiwanej wspólnej podsieci opartej o wierzchołki rozgałęziające.

9.1.2 Idea

Podział problemu na podproblemy, których rozwiązanie jest korzystne (możliwe) obliczeniowo w przeciwieństwie do bazowego problemu, nie jest nowym pomysłem w teorii grafów [126, 20] jak i szeroko pojętej nauce (np. algorytm Cooleya-Tukeya). W kontekście porównywania grafów możliwość działania na mniejszych strukturach jest preferowana ze względu na relatywnie dużą złożoność obliczeniową

wielu dostępnych algorytmów. Z tego powodu zastosowanie dekompozycji do porównania sieci Petriego uznano za dobry punkt wyjściowy. Należy jednocześnie pamiętać, że podział na podsieci nie jest bezstratny w kontekście informacji zawartych w bazowej sieci. Aby uzyskać rozwiązanie (wspólny podgraf) dla porównanych niedekomponowanych sieci, należy wykonać operację kompozycji (operację odwrotną do dekompozycji) na bazie znalezionych wspólnych podsieci. W tym momencie należy zadać kilka pytań w celu określenia praktyczności tego podejścia dla modeli sieci biologicznych wykonanych za pomocą sieci Petriego.

Czy złożoność obliczeniowa dekompozycji jest problemem? – podział sieci na podsieci ma zmniejszyć złożoność obliczeniową całego procesu porównywania i z tego powodu złożoność czasowa algorytmu dekompozycji powinna być liniowa lub wielomianowa. Analizując dostępne metody okazuje się, że nie dla wszystkich dekompozycji powszechnie stosowanych dla biologicznych sieci istnieją wielomianowe algorytmy ich wyznaczania. Wyznaczanie podsieci indukowanych przez zbiory MCT i conADT jest proste obliczeniowo, natomiast wyznaczanie zbioru t -niezmienników, na których bazują, jest zależne o rozmiaru i struktury sieci, o czym była mowa w Rozdziale 7. Istnieją jednak alternatywy, jak t -sieć, blisko powiązane z wymienionymi dekompozycjami, które nie są zależne od wcześniejszego wyznaczenia zbioru minimalnych niezmienników. Dokładna relacja została przedstawiona przy okazji opisu wierzchołków rozgałęziających w Rozdziale 5.

Czy istnieją metody kompozycji? – w dostępnej literaturze temat kompozycji wcześniej zdekomponowanych sieci Petriego nie był eksplorowany dla typów podsieci, na których operuje przedstawiony w tym rozdziale algorytm porównywania sieci. Jednocześnie, zbudowanie oryginalnej sieci bazując na etykietowanych podsieciach, które pokrywają się tylko na zbiorze miejsc rozgałęziających, jest prostą operacją.

Czy koniecznym jest przeprowadzenie kompozycji na bazie porównanych podsieci? – nie. Możliwość uzyskania największego wspólnego podgrafu jest znaczącą wartością dodatnią dla analizy sieci. Jednocześnie za dekompozycjami takimi jak np. opartym na zbiorach conADT leży potrzeba operowania na możliwie najmniejszych podsieciach reprezentujących swoją strukturą istotną informację. Porównanie takich podsieci dostarcza dane częściowo przetworzone do formy pożądanej w biologicznym kontekście. W ramach prezentowanych analiz nacisk stawiany był głównie na porównania i interpretacje wyników z perspektywy podsieci, a dopiero później całego modelu. Umotywowane jest to podziałem do fragmentów, które samodzielnie przedstawiają istotną informację.

9.2 Niematematyczne wyzwania

W trakcie rozwijania algorytmu przeprowadzono konfrontację zwracanych przez niego wyników z wiedzą ekspercką biologów (docelowych użytkowników), aby uzyskać uwagi dotyczące działania metody oraz wykryć braki, których modyfikacja była stosunkowo łatwa na danym etapie prac. W rezultacie udało się potwierdzić przewidywane trudności związane z zastosowaniem redukcji sieci przed właściwym procesem porównania, a także odkryto skomplikowaną kwestię dotyczącą tego, które elementy powinny znaleźć się we wspólnej podsieci, wartościowej z biologicznego punktu widzenia.

9.2.1 Redukcja sieci

W sekcji 2.2.3 przedstawione zostały redukcje struktury sieci Petriego, w wyniku przeprowadzenia których pomimo uproszczenia model pozostaje niezmienny z perspektywy matematycznej. To stwierdzenie jest prawdziwe, jeśli spoglądamy na model tylko z perspektywy jednego typu niezmienników. Część z redukcji bowiem wpływa na występowanie jednego typu niezmienników, jednocześnie nie zmieniając nic w liczbie przeciwnego typu. Jest to istotne z tego powodu, że w analizach wykonywanych na sieciach Petriego powszechnym jest skupianie się na jednym typie niezmienników (i metod bazujących na nich) z pominięciem analizy drugiego typu. W niektórych przypadkach może to być spowodowane nieistotnością danego typu niezmienników w badanej sieci.

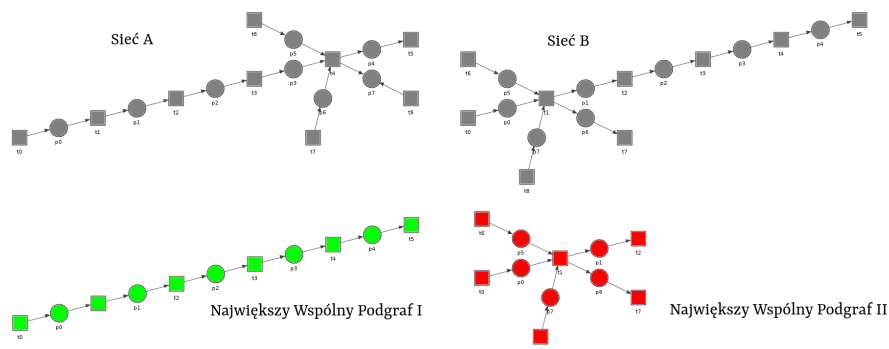
Część środowiska, z którą przeprowadzono konsultacje na temat kwestii redukcji sieci jest zdania, że nawet prosta redukcja w postaci zmniejszenia długości ścieżki łączącej dwa wierzchołki rozgałęziające zaburza przedstawiane informacje, co może prowadzić do błędnych wniosków na podstawie wyników porównania. Jest to sytuacja, która ma największy wpływ podczas porównania dwóch procesów, które zostały przedstawione za pomocą sieci Petriego na tym samym poziomie abstrakcji (szczegółowości). Jednocześnie sama redukcja modeli jako narzędzia do analizy dużych sieci budziła zainteresowanie w tej samej grupie.

Rozwiązaniem, które pozwala na uwzględnienie uwag biologów, jest przeprowadzanie procesu porównania dwukrotnie, osobno na oryginalnych i zredukowanych modelach. Nie jest to rozwiązanie optymalne, ze względu na podwojenie uzyskanych wyników, a także konieczność dodatkowej analizy porównawczej w celu wykrycia różnic i określenia ich przyczyny. O ile w przypadku modeli dekomponowanych do niewielkiej liczby podsieci jest to sytuacja pożądana, ze względu na dodatkowe informacje na temat struktury modelu, to w przypadkach sieci dekomponowanych do licznych podsieci może to stanowić dodatkowe obciążenie o relatywnie niskiej wartości informacyjnej. Mimo przedstawionych obiekcji, sugerowanym przez autora podejściem jest przeprowadzenie porównania na obu wariantach modeli.

9.2.2 Problem największego wspólnego podgrafu – biologiczna perspektywa

Szukając stopnia podobieństwa pomiędzy dwoma porównywanymi podsieciami, intuicyjnym jest wyznaczenie największej wspólnej struktury, jaka istnieje w obu przypadkach. Tutaj jednak pojawia się problem, ponieważ nie zawsze największa wspólna podsieć będzie tą, która zawiera najwięcej informacji kluczowych z perspektywy biologicznej. Prostym przykładem będzie największy wspólny podgraf o strukturze pojedynczej ścieżki, będącej rezultatem porównania dwóch drzew o korzeniach wysokiego stopnia – przykład na Rysunku 9.1.

W takiej sytuacji należy określić, jakie warunki musi spełniać podsieć, aby była uznana za istotną z biologicznej perspektywy. Jako punkt odniesienia wykorzystany został zbiór conADT, który był uznawany za najmniejszą podsieć o potencjalnym biologicznym znaczeniu. Analizując struktury, jakie mogą być indukowane przez zbiory tego typu, łatwo zauważyć, pomijając przypadki trywialne, że zbudowane są one na bazie wierzchołków rozgałęziających, gdzie rdzeń stanowią tranzycje rozgałęziające, z których wychodzą ścieżki ograniczone miejscami rozgałęziającymi – relacja dokładnie opisana w Rozdziale 5. Z tego powodu wspólna podsieć powinna zawierać możliwie największą liczbę wspólnych wierzchołków rozgałęziających wraz z połączeniami występującymi pomiędzy nimi. W efekcie preferowane są podsieci o strukturze drzewiastej nad prostymi ścieżkami.



Rysunek 9.1: Na dwóch sieciach przedstawiono dwa przypadki największych wspólnych podgrafów o takim samym rozmiarze, ale różnej wartości z perspektywy biologicznej. Zielonym kolorem zaznaczono podsieć o formie prostej ścieżki. Czerwonym kolorem zaznaczono podsieć opartą o tranzycję rozgałęziającą.

Niestety koncepcja opisana i opublikowana w pierwszym artykule proponującym heurystyczne podejście okazała się niewystarczająca. W ramach wspomnianych wcześniej konsultacji ujawniły się liczne przypadki, w których oczekiwano innej podsieci niż zwrócona. Sieci Petriego pozwalają modelować różnorodne procesy biologiczne (i nie tylko) często na różnych poziomach szczegółowości. Dodatkowo nie opracowano powszechnie uznawanego zestawu zasad określających styl w jakim powinny być tworzone sieci reprezentujące systemy biologiczne. Dla szeroko pojętych modeli układów biologicznych zbiór takich zasad byłby zbyt ogólny. Autorzy modeli operują też na różnych poziomach szczegółowości, czasem w ramach jednej sieci, co dodatkowo utrudnia analizę. Struktura, która będzie uznana za wspólną przy porównaniu modelu A z B, nie będzie za taką uznawana przy porównaniu B z C, ze względu na kontekst w jakim zostały utworzone. Efektem takiej sytuacji jest brak możliwości narzucenia ograniczeń, które pozwalałyby na sztywną interpretację struktur i dały wygodę w optymalizacji algorytmu.

Jednocześnie w wyniku przeprowadzonych konsultacji udało się określić częste przypadki, w których interpretacja bywa różna w zależności od kontekstu i dostosować algorytm do wyszukiwania pożądanych struktur. Rezultatem są cztery parametry sterujące formą uzyskanej wspólnej podsieci, opisane w sekcji 9.3.3.

9.2.3 Wspólna podsieć

Wspólna podsieć budowana na bazie zbiorów ADT i t-sieci będzie ograniczona przez zasady, jakim one podlegają. Podsieć taką można zdekomponować do zbioru ścieżek łączących dwa wierzchołki rozgałęziające, które występują tylko na pierwszej i ostatniej pozycji w ścieżce. Rozróżniamy trzy typy takich ścieżek:

- t-t - występująca pomiędzy tranzycjami rozgałęziającymi;
- t-p - występująca pomiędzy tranzycją a miejscem rozgałęziającym;
- p-p - występująca pomiędzy dwoma miejscami rozgałęziającymi.

Ścieżka t-t występuje w nietrywialnych podsieciach, zawierających przynajmniej dwie tranzycje rozgałęziające lub tranzycję rozgałęziającą i przynajmniej jedną tranzycję wejściową lub wyjściową o stopniu wierzchołka równym jeden.

Następnym przypadkiem jest ścieżka t-p, która występuje w podsieciach zawierających przynajmniej jedną tranzycję rozgałęziającą lub jedną tranzycję wejściową lub wyjściową o stopniu wierz-

chołka równym jeden. Obrazuje ona połączenie pomiędzy tranzycją a miejscem rozgałęziających (w przypadku t-sieci zawsze granicznym).

Ostatni przypadek, czyli p-p, odpowiada samodzielnej trywialnej podsieci o formie ścieżki.

Na Rysunkach 9.6 i 9.7 przedstawiono rozmieszczenie poszczególnych typów ścieżek w kilku podsieciach typowych dla t-sieci i indukowanych przez zbiory ADT.

Wynikiem porównania dwóch podsieci jest wspólna podsieć o maksymalizowanej liczbie zmapowanych wierzchołków rozgałęziających, połączonych ze sobą ścieżkami t-t, ze ścieżkami t-p połączonymi z pojedynczą tranzycją rozgałęziającą.

Zaznaczyć należy, że porównywane są struktury podsieci bez uwzględniania etykiet przypisanych do każdego wierzchołka. Jest to spowodowane brakiem znormalizowanego nazewnictwa reakcji w porównywanych sieciach. Wyjątkiem będą modele tworzone w oparciu o źródło danych narzucające standard etykiet, jak np. bazy danych stosujące klasyfikację Enzyme Commission number wspomnianą w Rozdziale 7. Ta sytuacja wymusiła skupienie się na strukturach, jakie współdzielą porównywane sieci. Jednocześnie modyfikacja algorytmu, aby uwzględniał kompatybilność etykiet, jest prosta do wprowadzenia w przypadku potencjalnej zmiany trendów w modelach wyrażonych za pomocą sieci Petriego.

Różnica pomiędzy sieciami indukowanymi przez zbiór ADT a t-sieciami

W przypadku podsieci indukowanych przez zbiory ADT, zbiór miejsc rozgałęziających nie musi odpowiadać zbiorowi miejsc granicznych. W przypadku, gdy model nie spełnia własności nie blokowalności (ang. non-blocking – dla każdego miejsca suma wag łuków wejściowych jest równa sumie wag łuków wyjściowych), może zaistnieć sytuacja, gdzie w sieci indukowanej przez zbiór ADT pojawi się miejsce rozgałęziające niebędące punktem styku z innym zbiorem ADT – przykład na Rysunku 9.2. Wszystkie tranzycje wchodzące i wychodzące z danego miejsca będą należeć do wsparć tych samych niezmienników, a zatem będą należeć do tej samej sieci indukowanej przez zbiór ADT. Ten sam fragment sieci będzie zdekomponowany do 2 podsieci t-sieci. Dla podsieci tego typu zbiory miejsc granicznych i rozgałęziających są tożsame.



Rysunek 9.2: Zaprezentowano przypadek podziału jednej podsieci indukowanej przez zbiór conADT na dwie podsieci t-sieci, gdzie podział zachodzi w miejscu rozgałęziającym (zaznaczonym szarym kolorem na prawej sieci).

Różnica ta wymogła niewielkie zmiany w działaniu algorytmu dla podsieci indukowanych przez zbiory conADT i MCT.

9.3 Proces porównywania podsieci

Rdzeniem dla opracowanej metody jest wyznaczenie dla dwóch porównywanych podsieci największych wspólnych podsieci opartych o wierzchołki rozgałęziające o własnościach opisanych w poprzednich sekcjach. Uzyskane podsieci w wyniku mapowania wierzchołków nie muszą tworzyć spójnej struktury. Jako że spójne podsieci są preferowane dla większości rozwiązań, jest to domyślny wariant sieci zwracany przez algorytm.

9.3.1 Format wyniku

Sumarycznie porównanie podsieci jest wykonywane $|n| \cdot |m|$ razy, gdzie n oznacza zbiór zdekomponowanych podsieci z sieci A , a m analogiczny zbiór dla sieci B . Jako uzupełnienie uzyskanych wyników porównania sugerowane jest sprawdzenie wewnętrznego podobieństwa podsieci w ramach oryginalnej sieci, czyli dodatkowe $(|n| - 1) \cdot (|n| - 1)$ i $(|m| - 1) \cdot (|m| - 1)$ porównań. Wykonanie porównania pomiędzy podsieciami należącymi do jednej sieci nie jest niezbędnym krokiem, jednak uzyskane za jego pomocą informacje mogą wpłynąć pozytywnie na jakość interpretacji wyników porównywania.

Rezultat porównania przedstawiany jest w postaci macierzy wyników cząstkowych $result^n$, gdzie każda pozycja w macierzy $result^n[i][j]$, odpowiada wartości funkcji $f^n(sub_1^A, sub_1^B)$:

$$f^n(sub_i^A, sub_j^B) = \frac{N^{sub-i-j}}{N^{max}}, \quad (9.1)$$

gdzie $|N^{sub-i-j}|$ oznacza liczbę wierzchołków dla wspólnej podsieci $sub-i-j$, będącej wynikiem porównania podsieci $sub-i$ z sieci A i podsieci $sub-j$ z sieci B .

$$result^n = \begin{bmatrix} f^n(sub_1^A, sub_1^B) & \dots & f^n(sub_1^A, sub_j^B) \\ \vdots & \ddots & \vdots \\ f^n(sub_i^A, sub_1^B) & \dots & f^n(sub_i^A, sub_j^B) \end{bmatrix} \quad (9.2)$$

Alternatywną formą jest użycie macierzy $result^a$ zawierającej wartości odpowiadające liczbie łuków znajdujących się w danych podsieciach, gdzie każda pozycja w $result^a[i][j]$, odpowiada wartości zwracanej przez funkcję $f^a(sub_1^A, sub_1^B)$:

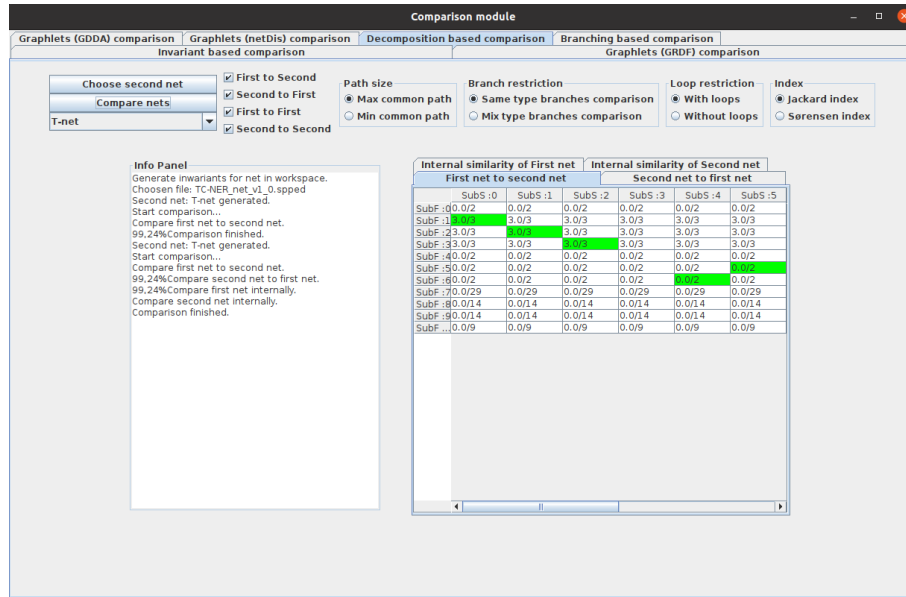
$$f^a(sub_i^A, sub_j^B) = \frac{A^{sub-i-j}}{A^{max}}, \quad (9.3)$$

gdzie $|A^{sub-i-j}|$ oznacza liczbę łuków we wspólnej podsieci $sub-i-j$, będącej wynikiem porównania podsieci $sub-i$ z sieci A i podsieci $sub-j$ z sieci B .

$$result^a = \begin{bmatrix} f^a(sub_1^A, sub_1^B) & \dots & f^a(sub_1^A, sub_j^B) \\ \vdots & \ddots & \vdots \\ f^a(sub_i^A, sub_1^B) & \dots & f^a(sub_i^A, sub_j^B) \end{bmatrix} \quad (9.4)$$

Przykład powyższej macierzy wynikowej wygenerowanej za pomocą aplikacji Holmes znajduje się na Rysunku 9.3.

W starszej wersji algorytmu znanej z publikacji [123], wynik cząstkowy przedstawiany był w postaci trójki $\{T, P, F\}$ przedstawiającej kolejno liczbę tranzycji, miejsc i łuków znajdujących się w podsieci.



Rysunek 9.3: Okno aplikacji Holmes z modułu odpowiedzialnego za porównywanie sieci. Wybrana zakładka przedstawia wyniki porównania podsieci t-sieć dla dwóch modeli.

Niestety przedstawienie w tej formie traci na czytelności w przypadku macierzy zbudowanej na kilkunastu podsieciach z każdej z porównywanych sieci. To spowodowało rozbitcie na dwie osobne macierze dla wierzchołków i łuków. Zrezygnowano również z rozróżnienia na typ wierzchołka.

9.3.2 Algorytm

Pierwszą operacją jest wyznaczenie podsieci za pomocą wybranej dekompozycji. W przypadku użycia podsieci bazujących na niezmiennikach konieczne będzie wcześniejsze wyznaczenie zbioru minimalnych t- lub p-niezmienników. Dekompozycję wykonujemy dla każdej z sieci uczestniczących w porównaniu (Algorytm 7). Następnie dla każdej pary (sub^A, sub^B) wyznaczana jest wspólna podsieć (Algorytm 8).

Etap 0 – Wyznaczenie tranzycji rozgałęziających

Dla każdej pary porównywanych podsieci (sub^A, sub^B) , wyznaczane są zbiory tranzycji rozgałęziających, jakie zawierają. W zależności od ich liczby struktury porównywanych podsieci dzielone są na trzy grupy ze względu na liczbę wierzchołków rozgałęziających jakie zawierają:

- *typ0* - przedstawia podsieć trywialną, posiadającą dokładnie jedną ścieżkę p-p;
- *typ1* - przedstawia podsieć o strukturze drzewiastej, posiadającą przynajmniej jedną ścieżkę t-p;
- *typb* - przedstawia podsieć dla rozbudowanej struktury, posiadającą przynajmniej jedną ścieżkę t-t.

W zależności od liczby rozgałęziających tranzycji proces porównania różni się. W przypadkach, gdzie występuje przynajmniej jedna podsieć typu 0 poszukiwana jest najdłuższa wspólna ścieżka rozpoczynając od miejsca rozgałęziającego. W przypadku, gdy występuje typ 1 i typ b, pojedyncza tranzycja z typ 1 jest sprawdzana z każdym możliwym dopasowaniem tranzycji rozgałęziającej z typ b – Etap III. Dla porównań typ b do typ b stosujemy pełną procedurę.

Algorytm 7 Porównanie zdekomponowanych sieci Petriego przez znalezienie wspólnych podsieci.

Input: PN_A, PN_B - dwie porównywane sieci Petriego
Input: d - flaga odpowiadająca za typ dekompozycji
Output: $result_{A-B}$ - macierz wynikowa podobieństw między PN_A a PN_B
Output: $result_{A-A}$ - macierz wynikowa z wewnętrznymi podobieństwami PN_A
Output: $result_{B-B}$ - macierz wynikowa z wewnętrznymi podobieństwami PN_B

```

1: procedure COMPARISONANDDECOMPOSITION( $PN_A, PN_B, d$ )
2:    $subnets_A = decompose(PN_A, d)$  ▷ Dekompozycja do typu  $d$ 
3:    $subnets_B = decompose(PN_B, d)$ 
4:    $result_{A-B} = comparison(subnets_A, subnets_B)$  ▷ Porównanie podsieci z  $PN_A$  i  $PN_B$ 
5:    $result_{A-A} = comparison(subnets_A, subnets_A)$  ▷ Wewnętrzne porównanie  $PN_A$ 
6:    $result_{B-B} = comparison(subnets_B, subnets_B)$  ▷ Wewnętrzne porównanie  $PN_B$ 
7: end procedure

1: procedure COMPARISON( $subnets_A, subnets_B$ )
2:    $result[][]$  ▷ Macierz wynikowa
3:   for all  $sub_i \in subnets_A$  do
4:     for all  $sub_j \in subnets_B$  do
5:        $subnets = findCommSubnets(sub_i, sub_j)$ 
6:        $common\_subnet\_size = getSizeOfMaximumElement(subnets)$ 
7:        $m = max(getSize(sub_i), getSize(sub_j))$ 
8:        $result[i][j] = common\_subnet\_size/m$ 
9:     end for
10:  end for
11:  return  $result$ 
12: end procedure

```

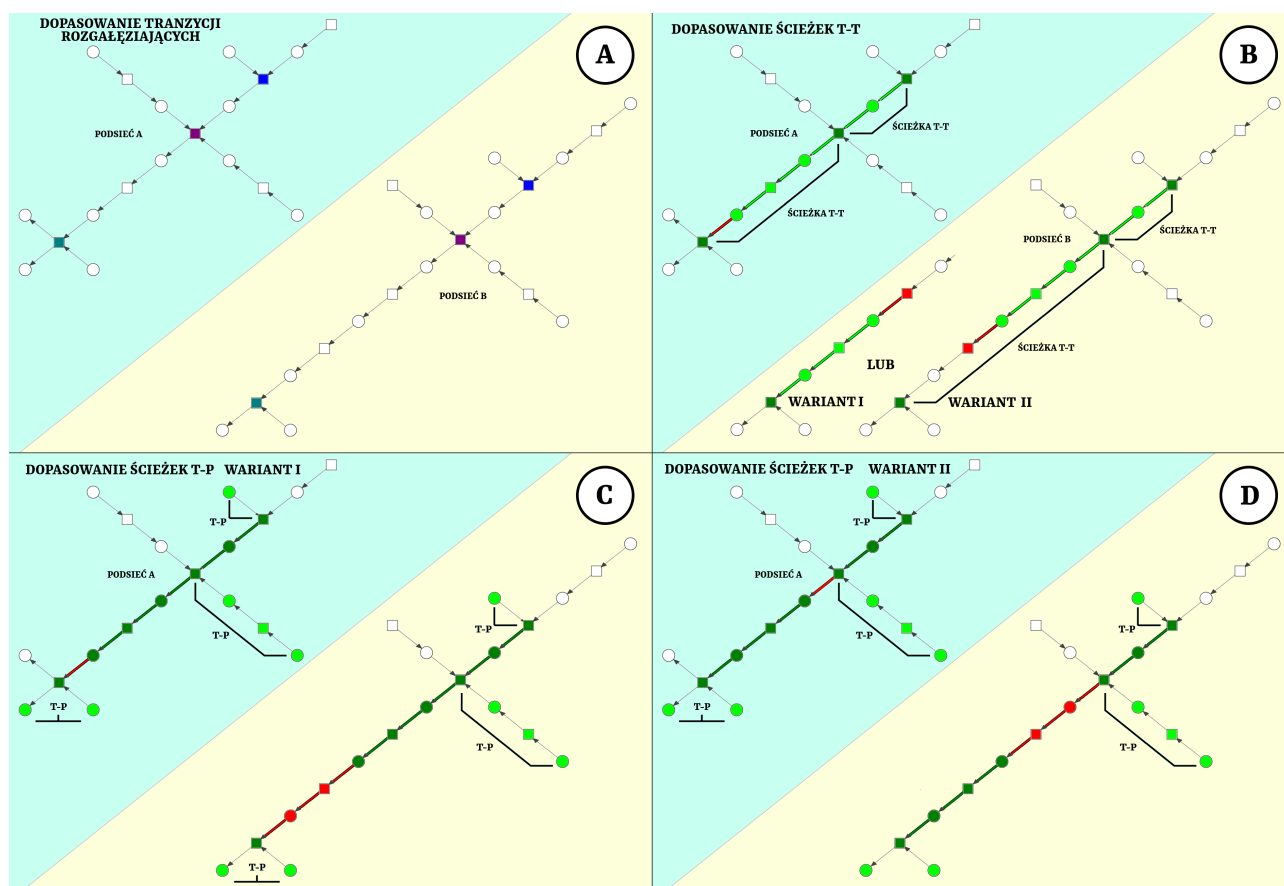
Algorytm 8 Wyznaczenie wspólnej podsieci dla pary zdekomponowanych sub_1 i sub_2

Input: sub_1 and sub_2 - para porównywanych podsieci
Input: $q1$ - flaga odpowiadająca za pytanie z sekcji 9.3.3
Output: $subnets$ - zbiór znalezionych wspólnych podsieci

```

1: procedure FINDCOMMONSUBNETS( $sub_1, sub_2$ )
2:    $matchedPaths = findMatching(sub_1, sub_2)$ 
3:   for all  $matching_i \in matchedPaths$  do
4:      $com = \emptyset$  ▷ Inicjalizacja wspólnej podsieci
5:     for all  $matching_i$  do
6:        $path = compareCorePaths(matching_i)$  ▷ znajdowanie wspólnej ścieżki t-t
7:        $com.add(path)$  ▷ dodawanie znalezionej ścieżki do wspólnej podsieci
8:     end for
9:     for all  $t_i^{br} \in com$  do
10:      if  $q1$  then ▷ Wybór wariantu wspólnej podsieci
11:         $path = compareBranchPaths(getTt - paths(t_i^{br}, sub_1, sub_2))$ 
12:         $com.add(path)$ 
13:      else
14:         $path = compareBranchPaths(getTTPaths(t_i^{br}, sub_1, sub_2))$ 
15:         $com.add(path)$ 
16:         $path = compareBranchPaths(gett - paths(t_i^{br}, sub_1, sub_2))$ 
17:         $com.add(path)$ 
18:      end if
19:    end for
20:     $subnets.add(com)$ 
21:  end for
22:   $subnets.removeNonMaximal()$  ▷ Usuwanie podsieci o rozmiarze mniejszym niż maksymalny znaleziony
23:  return  $subnets$ 
24: end procedure

```



Rysunek 9.4: Wizualizacja procesu znajdowania wspólnej podsieci. W części A za pomocą kolorów zaprezentowano pary dopasowanych wierzchołków. W części B przedstawione jest znajdowanie wspólnych ścieżek t-t występujących pomiędzy zmapowanymi wierzchołkami z części A. W przypadkach, gdy ścieżka t-t nie jest równej długości w obu porównywanych podsieciach, należy rozważyć budowę dwóch struktur, gdzie ścieżka ta będzie połączona z każdą z dwóch zmapowanych par wierzchołków. Dla rozważanych dwóch wariantów w częściach C i D przedstawiono kolejny krok, jakim jest znalezienie ścieżek t-p występujących jednocześnie w obu podsieciach.

Etap I – Dopasowanie tranzycji rozgałęziających

Poszukując największej wspólnej struktury należy rozważyć wszystkie możliwe mapowania tranzycji rozgałęziających pomiędzy porównywanymi sieciami. Przeszukanie całej przestrzeni rozwiązań ma złożoność czasową $b!$, gdzie b odpowiada liczbie tranzycji rozgałęziających występujących w mniejszej z dwóch porównywanych podsieci. W efekcie ze względu na liczbę mapowań, praktyczne jest przeszukiwanie całej przestrzeni dla przypadków gdzie podsieci zawierają do 12 tranzycji rozgałęziających. Podsieci zawierające tyle tranzycji rozgałęziających są rzadkie pośród modeli systemów biologicznych. Dodatkowo w zdecydowanej większości przypadków przeszukiwanie całej przestrzeni rozwiązań jest niepraktyczne.

Dlatego też prosta heurystyka ograniczająca liczbę sprawdzanych porównań została zaimplementowana. Ta kwestia przedstawia duży potencjał optymalizacyjny, jednak na potrzeby typowych podsieci występujących w modelach systemów biologicznych zawierających średnio 6 tranzycji rozgałęziających, zaimplementowane podejście jest wystarczające. Polega ono na wyznaczeniu połączeń (ścieżek lub bezpośrednich łuków) pomiędzy tranzycjami rozgałęziającymi. Do dalszego etapu przechodzą tylko mapowania przedstawiające połączone ze sobą tranzycje rozgałęziające ścieżkami o właściwym kierunku, które występują w obu porównywanych podsieciach – Rysunek 9.4, część A.

Etap II – Dopasowanie ścieżek t-t

Dla uzyskanych w poprzednim kroku mapowań tranzycji rozgałęziających sprawdzane są ścieżki t-t istniejące pomiędzy nimi. W przypadku równej długości w obu podsieciach sprawdzane są wagi łuków, wyznaczając największą wspólną wagę dla każdego łuku – Rysunek 9.4, część B. W przypadku, gdy ścieżki nie są równej długości, budowane są dwa przypadki, szukając najdłuższej ścieżki rozpoczynając od tranzycji startowej i osobno rozpoczynając od tranzycji końcowej. Podobnie jak w poprzednim przypadku porównywana jest waga łuków. Uzyskana podsieć bazująca na ścieżkach t-t przechodzi do kolejnego kroku.

Etap III – Dopasowanie ścieżek t-p

W poprzednim kroku utworzony został rdzeń podsieci bazujący na ścieżkach t-t łączących zmapowane tranzycje rozgałęziające. Nie oznacza to, że wszystkie ścieżki t-t występujące w obu podsieciach zostały użyte. Za to może odpowiadać różnica w liczbie tranzycji rozgałęziających oraz występowanie ścieżek t-t łączących tranzycje wyjściowe lub wejściowe.

Przechodząc pomiędzy wierzchołkami rozgałęziającymi występującymi w utworzonej już wspólnej strukturze dokonywane jest porównanie ścieżek t-p połączonych ze zmapowanymi wierzchołkami oraz nieużytych jeszcze ścieżek t-t – Rysunek 9.4, części C i D. W domyślnym wariancie porównanie ścieżek t-p i t-t jest przeprowadzane rozdzielnie ze względu na różnice w przepływie, jakie przedstawiają.

Rezultatem jest zbiór największych wspólnych struktur występujących dla każdego ze znalezionych mapowań. Znalezione podsieci mogą znacząco różnić się strukturą ze względu na pochodzenie z różnych mapowań.

9.3.3 Dostępne warianty wspólnych podsieci

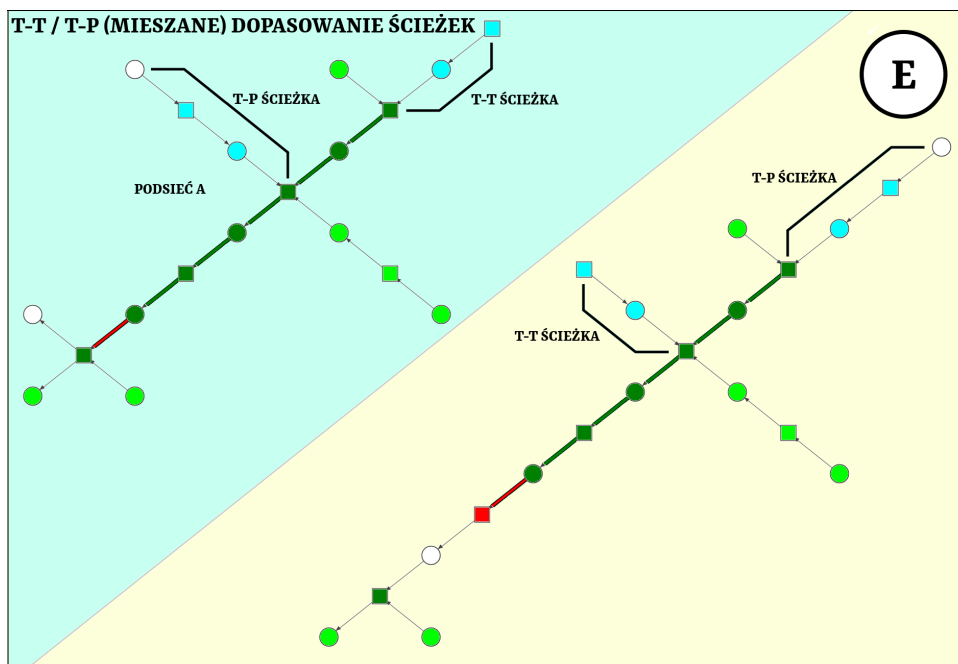
Opisane poniżej warianty wspólnych podsieci mogą być ze sobą łączone w celu uzyskania pożądaných struktur.

Problem samotnego wierzchołka – interpretacja stopni wierzchołków

Kiedy można stwierdzić, że dwie sieci są swoją strukturą całkowicie do siebie niepodobne? Czy porównując dwie trywialne podsieci o formie łuku łączącego tranzycję z miejscem, ale o przeciwnym zwrocie, częścią wspólną jest pojedyncza tranzycja lub miejsce? Czy też część wspólna nie istnieje, ponieważ stopnie wierzchołków się nie zgadzają – wchodzący i wychodzący stopień. Sytuacja ta implikuje kolejne pytanie. Czy mapowaniu jako podobne podlegają tylko wierzchołki tego samego stopnia wejściowego i wyjściowego, czy też odrzucamy idealne mapowanie i poszukujemy największego dostępnego mapowania? Ze względu na kontekst biologiczny domyślnym podejściem jest mniej restrykcyjna interpretacja, pozwalająca na zmapowanie wierzchołków o różnych stopniach.

Porównanie ścieżek różnego typu (t-t/t-p)

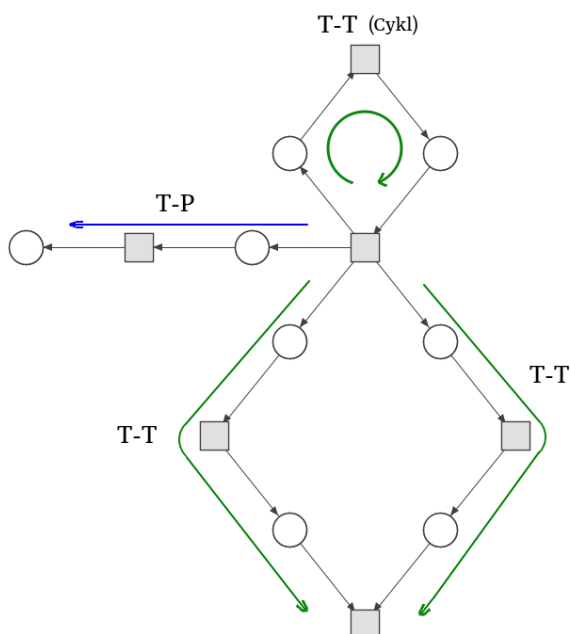
Przy opisie poszukiwanej wspólnej podsieci przedstawiony został podział na typy ścieżek wychodzących z tranzycji rozgałęziającej – t-t/t-p. Opisują one różne funkcje w ramach podsieci, w których występują. Z tego powodu podczas wyznaczania wspólnej podsieci porównywane są one w osobnych krokach. Istnieją jednak przypadki modeli, w których ze względu na kontekst, takie ograniczenie nie



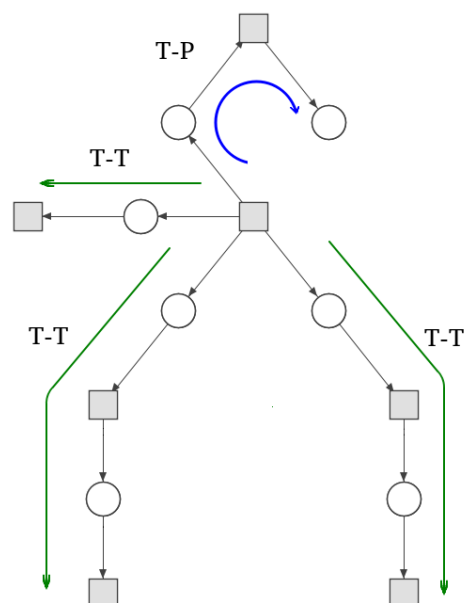
Rysunek 9.5: Krok E opisuje wariant algorytmu w którym dozwolone jest dopasowywanie niewykorzystanych w kroku B ścieżek t-t ze ścieżkami t-p. Kolorem niebieskim zaznaczono fragmenty sieci jakie zostały zmapowane przez algorytm przy tych ustawieniach.

jest preferowane. W rezultacie powstał wariant wspólnej podsieci powstały przez pozwolenie na przeprowadzenie dopasowań ścieżek z pominięciem ich typu, skupiając się na znalezieniu najdłuższych wspólnych ścieżek – Rysunek 9.5, część E.

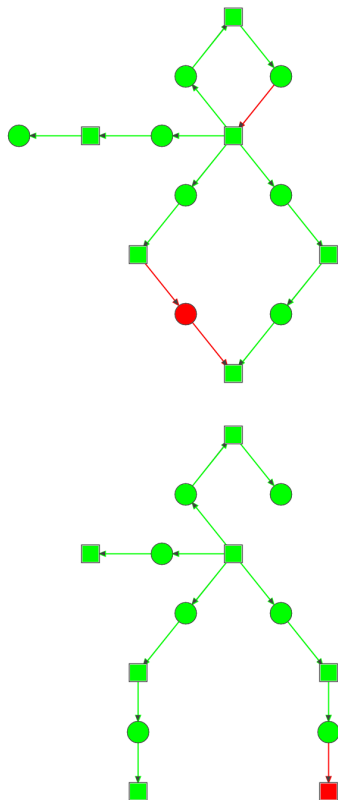
Jednocześnie należy zaznaczyć, że przy pozwoleniu na porównanie ścieżek różnego typu, przeprowadzenie później operacji kompozycji na bazie uzyskanych podsieci może dać niepoprawny rezultat.



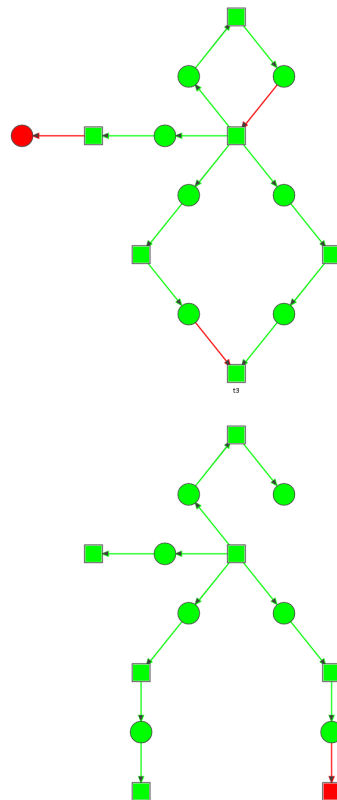
Rysunek 9.6: Przykładowa sieć zawierająca jedną ścieżkę t-t będącą jednocześnie cyklem, dwie ścieżki t-t i jedną ścieżkę t-p



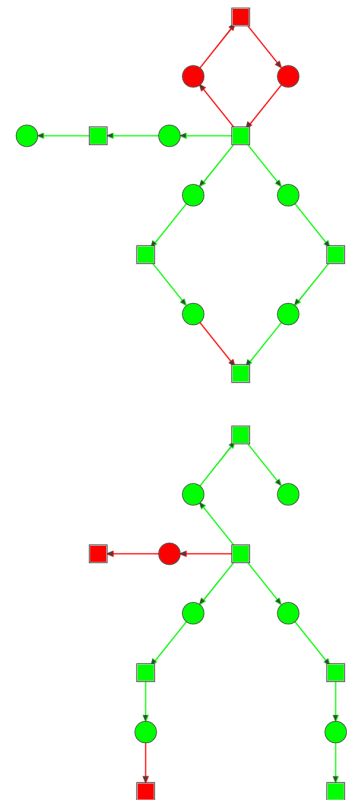
Rysunek 9.7: Przykładowa sieć zawierająca trzy ścieżki t-t i jedną ścieżkę t-p



Rysunek 9.8: Wspólna podsieć z pętlami, bez mieszanego dopasowania



Rysunek 9.9: Wspólna podsieć z pętlami i mieszanym dopasowaniem



Rysunek 9.10: Wspólna podsieć bez pętli i bez mieszanego dopasowania

Kwestia pętli

Pętle w sieciach Petriego są kolejnym przypadkiem, gdzie matematyczny zapis staje w pewnym konflikcie z interpretacją biologiczną. Usunięcie z modelu całej pętli zaczynającej się w tranzycji nie wpływa na liczbę t-niezmienników sieci (wpływa na liczbę p-niezmienników). Z tego powodu są przesłanki do traktowania pętli jako specyficznego przypadku i porównywania ich osobno względem innych ścieżek t-t.

Spójność

Ostatnią z rozważanych charakterystyk jest spójność poszukiwanej wspólnej podsieci. Domyślnie poszukiwane są spójne podsieci.

9.4 Przykład zastosowania

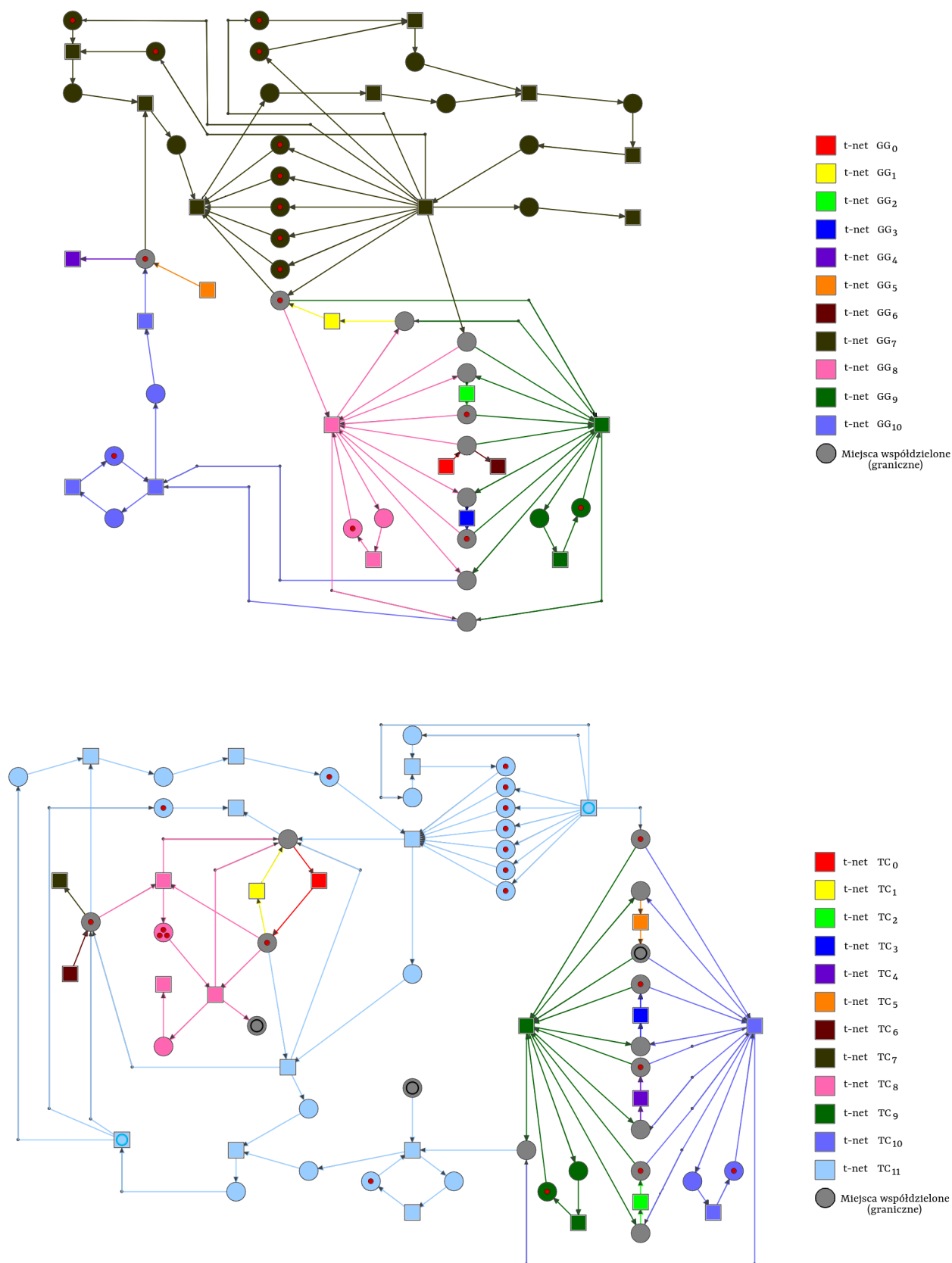
Przykład zastosowania metody porównywania zostanie przedstawiony na podstawie dwóch sieci opisujących procesy NER [111].

Opis biologiczny

Naprawa przez wycinanie nukleotydów (ang. Nucleotide excision repair (NER)) jest jedną z kilku ścieżek naprawy DNA w żywych komórkach ssaków. Uszkodzenia korygowane przez NER obejmują zmiany w DNA powstałe w wyniku działania promieniowania UV, mutagenów środowiskowych oraz niektórych chemoterapeutycznych adduktów z DNA. Naprawa przez wycinanie nukleotydów ma dwa podtypy: globalny genomowy NER (GG-NER górny model na Rysunku 9.11) i NER sprzężony z transkrypcją (TC-NER dolny model na Rysunku 9.11). TC-NER jest odpowiedzialny za przyspieszoną naprawę zmian w transkrybowanej nici aktywnych genów. GG-NER może zachodzić w dowolnym miejscu genomu. GG-NER jest inicjowany przez specyficzny dla GG-NER czynnik XPC-RAD23B, w niektórych przypadkach z pomocą UV-DDB (ang. UV-damaged DNA-binding protein). TC-NER jest inicjowane przez polimerazę RNA zatrzymaną w miejscu uszkodzenia za pomocą TC-NER specyficznych czynników CSA, CSB i XAB2. Oba szlaki wymagają do zakończenia procesu wycinania czynników rdzeniowych procesu NER [111, 100].

Wyniki cząstkowe porównania podsieci z modeli GG-NER i TC-NER zostały przedstawione na Tabelach 9.2, 9.1, 9.3. Dwie pierwsze przedstawiają rezultaty porównania wewnętrznego dla każdej z sieci. Izomorfizm został zaznaczony ciemnozielonym kolorem, podczas gdy zawieranie się sieci w pełni w innej podsieci zaznaczono używając koloru jasnozielonego. Czerwień sygnalizuje brak istotnych wspólnych elementów pomiędzy podsieciami (np. przepływowa podsieć i rozgałęziająca domyślnie uznawane są za niepodobne, co ma podstawy w kontekście znacznej części modeli biologicznych). Pozostałe pośrednie przypadki zostały pozostawione bez dodatkowych oznaczeń.

Na wynikach z Tabeli 9.3, używając metody węgierskiej, przeprowadzono wyszukiwanie zbioru o największej sumarycznej wielkości (rezultat zaznaczony kolorem niebieskim). Zależności wynikające z nich zostaną przedstawione na przykładach w następnej sekcji.



Rysunek 9.11: Modele przedstawiają dwa podtypy mechanizmów naprawy DNA przez wycinanie nukleotydów. Pierwszym jest globalny genomowy podtyp naprawy DNA – GG-NER. Drugim sprzężony z transkrypcją podtyp naprawy DNA – TC-NER

Tabela 9.1: Wyniki porównania wewnętrznego podobieństwa dla TC-NER. Przedstawiają rozmiar największych wspólnych podsieci występujących pomiędzy podsieciami od TC_0 do TC_{11}

	TC_0	TC_1	TC_2	TC_3	TC_4	TC_5	TC_6	TC_7	TC_8	TC_9	TC_{10}	TC_{11}
TC_0	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3
TC_1	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3
TC_2	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3
TC_3	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3
TC_4	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3
TC_5	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3
TC_6	0/2	0/2	0/2	0/2	0/2	0/2	2/2	1/2	2/2	2/2	2/2	2/2
TC_7	0/2	0/2	0/2	0/2	0/2	0/2	1/2	2/2	2/2	2/2	2/2	2/2
TC_8	0/9	0/9	0/9	0/9	0/9	0/9	2/9	2/9	9/9	6/9	6/9	6/9
TC_9	0/14	0/14	0/14	0/14	0/14	0/14	2/14	2/14	6/14	14/14	14/14	6/14
TC_{10}	0/14	0/14	0/14	0/14	0/14	0/14	2/14	2/14	6/14	14/14	14/14	6/14
TC_{11}	0/35	0/35	0/35	0/35	0/35	0/35	2/35	2/35	6/35	6/35	6/35	35/35

Tabela 9.2: Wyniki porównania wewnętrznego podobieństwa dla GG-NER. Przedstawiają rozmiar największych wspólnych podsieci występujących pomiędzy podsieciami od GG_0 do GG_{10}

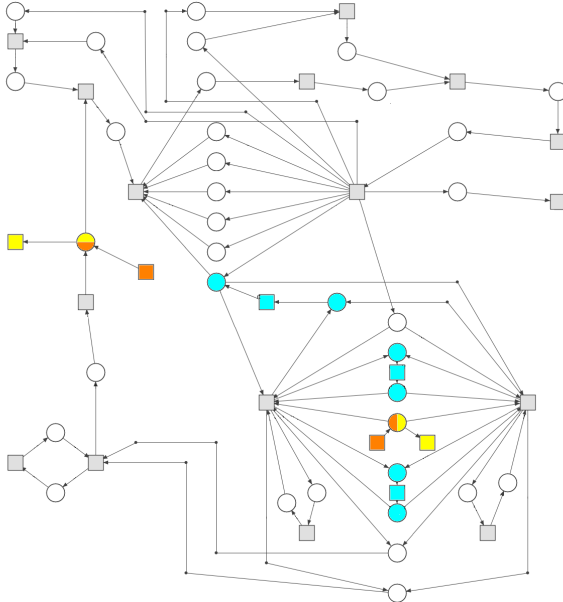
	GG_0	GG_1	GG_2	GG_3	GG_4	GG_5	GG_6	GG_7	GG_8	GG_9	GG_{10}
GG_0	2/2	0/2	0/2	0/2	1/2	2/2	1/2	2/2	2/2	2/2	2/2
GG_1	0/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
GG_2	0/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
GG_3	0/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
GG_4	1/2	0/2	0/2	0/2	2/2	1/2	2/2	2/2	2/2	2/2	2/2
GG_5	2/2	0/2	0/2	0/2	1/2	2/2	1/2	2/2	2/2	2/2	2/2
GG_6	1/2	0/2	0/2	0/2	2/2	1/2	2/2	2/2	2/2	2/2	2/2
GG_7	2/29	0/29	0/29	0/29	2/29	2/29	2/29	29/29	5/29	5/29	5/29
GG_8	2/14	0/14	0/14	0/14	2/14	2/14	2/14	5/14	14/14	14/14	7/14
GG_9	2/14	0/14	0/14	0/14	2/14	2/14	2/14	5/14	14/14	14/14	7/14
GG_{10}	2/9	0/9	0/9	0/9	2/9	2/9	2/9	5/9	7/9	7/9	9/9

Tabela 9.3: Wyniki porównania podobieństwa GG-NER do TC-NER

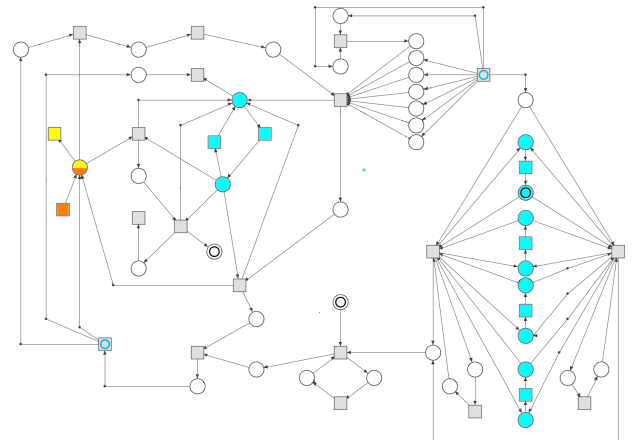
	TC_0	TC_1	TC_2	TC_3	TC_4	TC_5	TC_6	TC_7	TC_8	TC_9	TC_{10}	TC_{11}
GG_0	0/2	0/2	0/2	0/2	0/2	0/2	2/2	1/2	2/2	2/2	2/2	2/2
GG_1	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3
GG_2	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3
GG_3	3/3	3/3	3/3	3/3	3/3	3/3	0/3	0/3	0/3	0/3	0/3	0/3
GG_4	0/2	0/2	0/2	0/2	0/2	0/2	1/2	2/2	2/2	2/2	2/2	2/2
GG_5	0/2	0/2	0/2	0/2	0/2	0/2	2/2	1/2	2/2	2/2	2/2	2/2
GG_6	0/2	0/2	0/2	0/2	0/2	0/2	1/2	2/2	2/2	2/2	2/2	2/2
GG_7	0/29	0/29	0/29	0/29	0/29	0/29	2/29	2/29	6/29	5/29	5/29	19/29
GG_8	0/14	0/14	0/14	0/14	0/14	0/14	2/14	2/14	6/14	14/14	14/14	6/14
GG_9	0/14	0/14	0/14	0/14	0/14	0/14	2/14	2/14	6/14	14/14	14/14	6/14
GG_{10}	0/9	0/9	0/9	0/9	0/9	0/9	2/9	2/9	6/9	7/9	7/9	6/9

9.4.1 Przypadki trywialne

Przez przypadki trywialne rozumiemy podsieci t-sieć/ADT nie posiadające tranzycji rozgałęziającej. W efekcie mają one formę ścieżki, która występuje pomiędzy dwoma miejscami rozgałęziającymi (podsieci zaznaczone kolorem niebieskim na Rysunkach 9.12 i 9.13). Innymi przypadkami uznawanymi za sieci trywialne są podsieci indukowane przez jeden łuk łączący miejsce graniczne z tranzycją wyjściową (kolor żółty) lub tranzycję wejściową z miejscem granicznym (kolor pomarańczowy).



Rysunek 9.12: Trywialne podsieci w modelu GG-NER

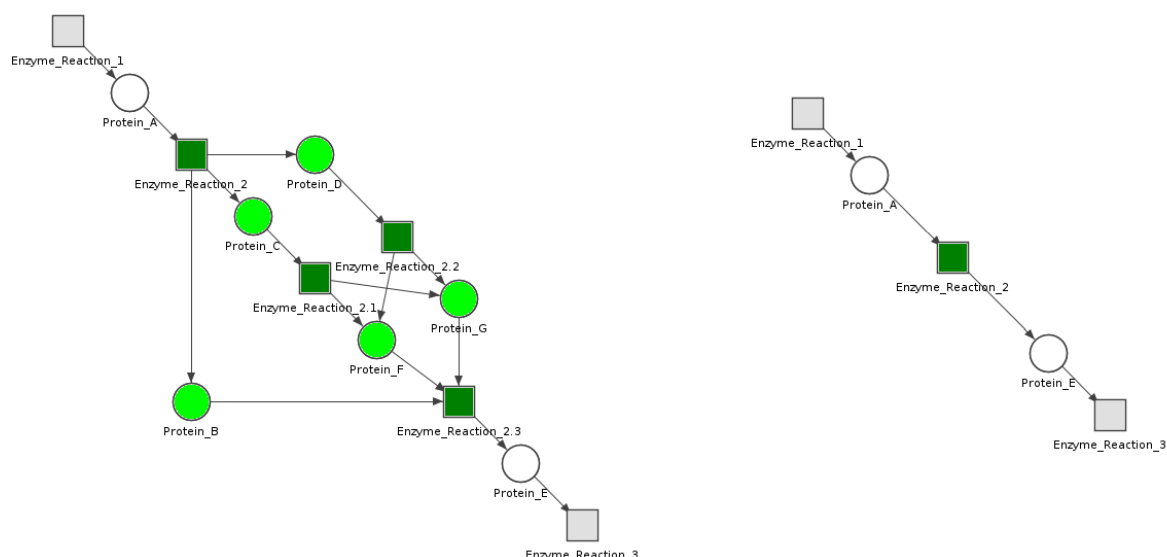


Rysunek 9.13: Trywialne podsieci w modelu TC-NER

Czy niewielki rozmiar trywialnych podsieci definiuje, że pełnią one nieistotną rolę w sieci? Niekoniecznie. Jest to zależne od szczegółowości, jaką reprezentuje model, ponieważ pojedyncza tranzycja przepływowa może odpowiadać jednej reakcji lub może przedstawiać skomplikowany podproces, wyrażony w uproszczonej formie – przykład na Rysunku 9.14. Z tego powodu możliwa jest sytuacja, gdzie w porównywanych modelach występuje ten sam proces, ale zamodelowany na różnych poziomach szczegółowości nie będzie posiadał podobnej struktury. Z tego powodu konieczne jest sprowadzenie modeli do zbliżonej szczegółowości, np. przez redukcję jednej z sieci. W takich przypadkach główną informacją zawartą w tranzycji przepływowej nie jest struktura, jaką indukuje, ale obszary/podprocesy, jakie łączy – więcej w sekcji 9.4.4.

Ze względu na częstą obecność dużej liczby trywialnych podsieci w porównywanych modelach uzyskana zostanie duża liczba dopasowań. Z perspektywy reprezentowanych struktur dopasowania te będą równie dobre, podczas gdy z perspektywy biologicznej będą reprezentować odmienne podprocesy. W Tabelach 9.2 i 9.1, widoczne jest, że kolejno podsieci od GG_1 do GG_3 i od TC_0 do TC_5 odpowiadają trywialnym podsieciom przepływowym, co powoduje, że w Tabeli 9.3 dopasowania w przedziale $GG_1 : GG_3$ a $TC_0 : TC_5$ są równie dobre z matematycznego punktu widzenia. Do eliminacji części niepoprawnych dopasowań można użyć etykiet i ich porównania.

Z wymienionych powyżej powodów możemy wydzielić dwa podejścia analizy podsieci trywialnych. Pierwsze podejście zakłada, że wszystkie podsieci trywialne znajdujące się w modelu powinny być analizowane zbiorczo, jako grupa prostych struktur łączących istotniejsze pod względem rozmiaru fragmenty modeli. Drugim jest analiza sąsiedztwa każdej z nich, co zostanie przedstawione w sekcji



Rysunek 9.14: Przykład przedstawiający dwa fragmenty sieci reprezentujące ten sam proces modelowany na różnych poziomach szczegółowości. Fragment sieci po lewej stronie zaznaczony zielonym kolorem w wyniku redukcji może zostać sprowadzony do postaci pojedynczej tranzycji widocznej w sieci po prawej stronie.

9.4.3.

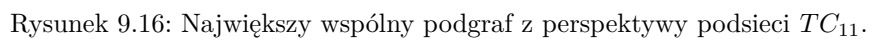
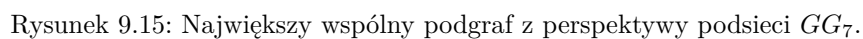
9.4.2 Częściowe dopasowanie

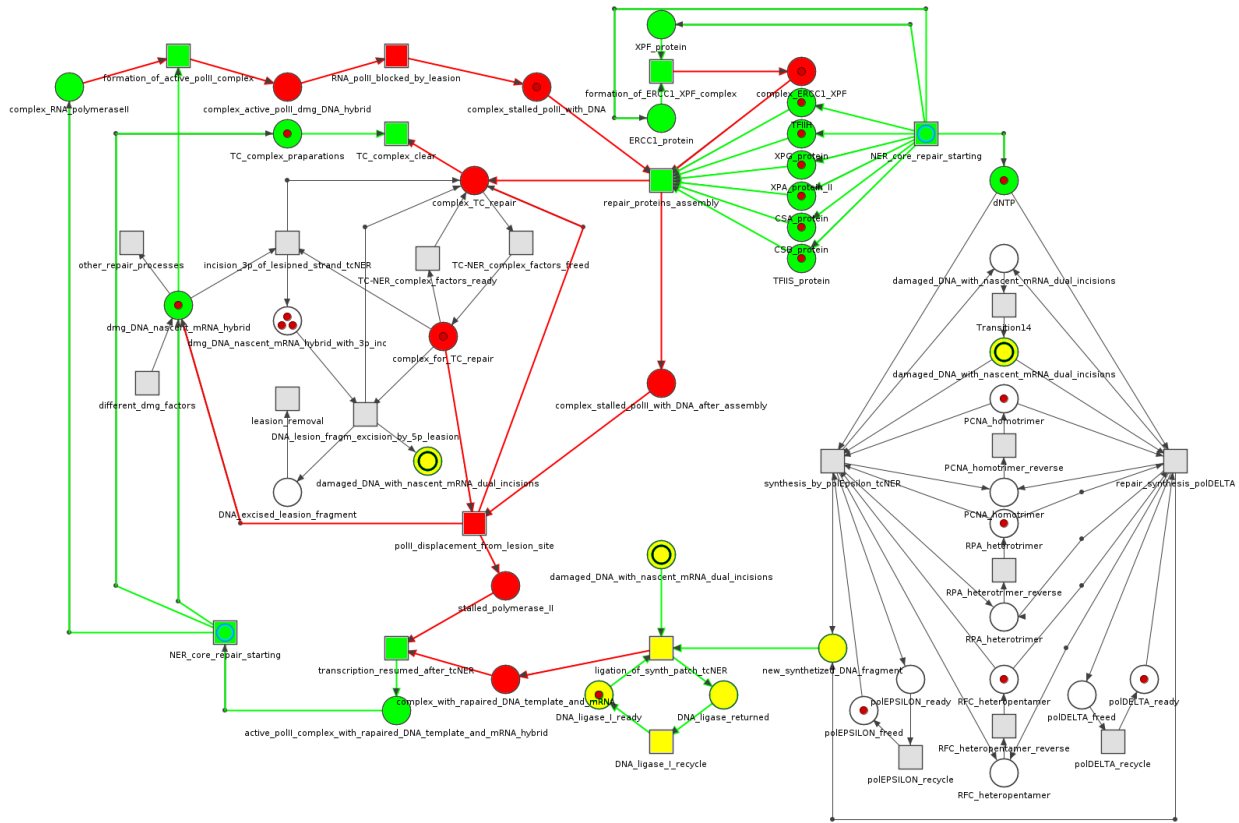
Sytuacja, gdy wspólna struktura znaleziona pomiędzy dwoma podsieciami reprezentuje jedynie ich fragment, jest najczęściej spotykanym rezultatem porównania. Jednocześnie jest to szeroki termin, pod którym skrywają się przypadki o różnym stopniu pożądania przez użytkownika:

- szczątkowych wspólnych struktur, np. w postaci pojedynczych wierzchołków pomiędzy kilkunastowierzchołkowymi sieciami, co samo w sobie niesie informację o braku strukturalnego podobieństwa, która może być pożądana w konkretnych przypadkach;
- niemal pełnego dopasowania, w którym udało się uzyskać wspólną strukturę o wielkości przekraczającej 70 – 80% rozmiaru porównywanych podsieci. Jest to przypadek przeciwny do wspomnianego w poprzednim punkcie;
- przypadek pośredni pomiędzy szczątkowym a pełnym dopasowaniem, obrazujący często przypadkowe strukturalne podobieństwo pomiędzy niepowiązanymi procesami;
- jednostronne zawieranie, które może zajść pomiędzy podsieciami o różnych rozmiarach i jednocześnie mogące obrazować każdy ze wspomnianych trzech przypadków porównania struktur.

Należy zaznaczyć, że nawet wysokie podobieństwo struktur uzyskane w wyniku porównania jest przesłanką, która musi zostać przeanalizowana w celu jej potwierdzenia. Na sytuację, gdzie częściowe dopasowanie jest interpretowane jako pożądane/wartościowe wpływa kilka czynników. Oczywistym jest rozmiar wspólnej podsieci, tj. liczba dopasowanych wspólnych wierzchołków rozgałęziających. Są to jednak wstępne warunki, jakie muszą spełnić podsieci nim zostaną odpowiednio zinterpretowane.

Przypadek prezentowany na Rysunkach 9.15 i 9.16 odpowiada porównaniu dwóch największych podsieci t-sieci występujących w sieciach GG-NER i TC-NER (kolorem zielonym zaznaczony jest





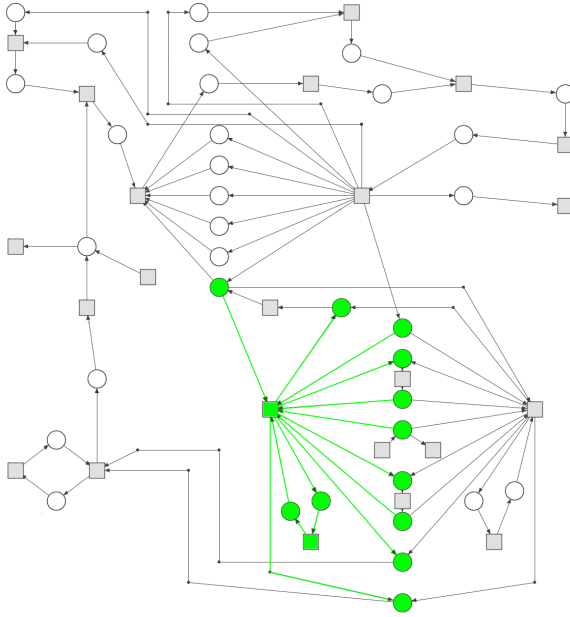
Rysunek 9.17: Przykład zawierania się dwóch podsieci GG_{10} (żółty) i GG_7 (zielony) w podsieci TC_{11} .

fragment wspólnej podsieci zmapowanej w obu sieciach, a czerwonym zaznaczone są obszary podsieci, które nie znalazły swoich odpowiedników). Rezultatem jest wspólna podsieć reprezentująca spójną strukturę stanowiącą kolejno około 0,65% GG_7 i 0,54% TC_{11} .

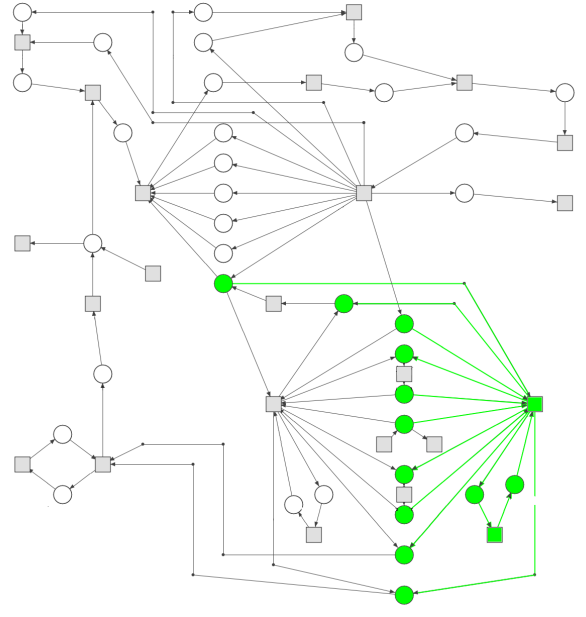
Warto zauważyć, że podsieć występująca w jednym modelu, może odpowiadać kilku sąsiadującym podsięciom w drugim. Przypadek takiej relacji został przedstawiony na Rysunku 9.17, pomiędzy podsięciami GG_{10} i GG_7 w podsieci TC_{11} . W części podsieci TC_{11} która nie została zmapowana z podsięcią GG_7 , zmapowano podsieć GG_{10} . Tranzycja rozgałęziająca wraz z cyklem stanowi charakterystyczną strukturę, która występuje w obu modelach. Dodatkowo w GG podsieci GG_{10} i GG_7 są ze sobą styczne przez miejsce graniczne. W przypadku usunięcia tranzycji wyjściowych i wejściowych, które są połączone z tym miejscem oraz skróceniu ścieżki łączącej teraz bezpośrednio tranzycje rozgałęziające, nastąpiłoby połączenie GG_{10} i GG_7 w jedną podsieć. W efekcie część wspólna pomiędzy podsięciami osiągnęłaby rozmiar około 0,77%.

9.4.3 Pełne dopasowanie

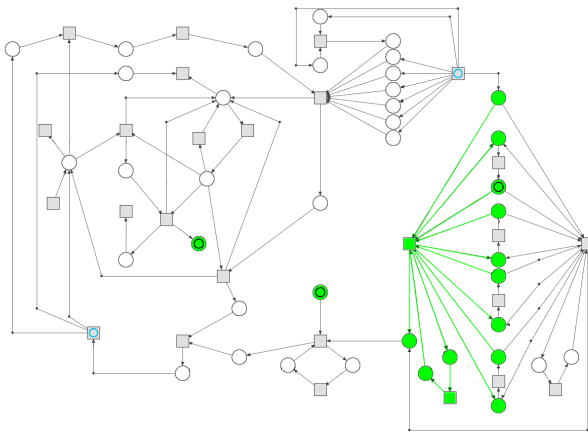
Ostatni z rozważanych przypadków dotyczy pełnego zmapowania dwóch porównywanych nietrywialnych struktur. Oznacza to, że pomiędzy podsięciami zachodzi izomorfizm. Jednocześnie nie oznacza, że porównane podsieci mają tę samą funkcję w modelach, z których pochodzą, a że współdzielą strukturę. Do potwierdzenia, że obie podsieci opisują identyczne lub bardzo podobne podprocesy jest konieczne użycie wiedzy specjalistycznej.



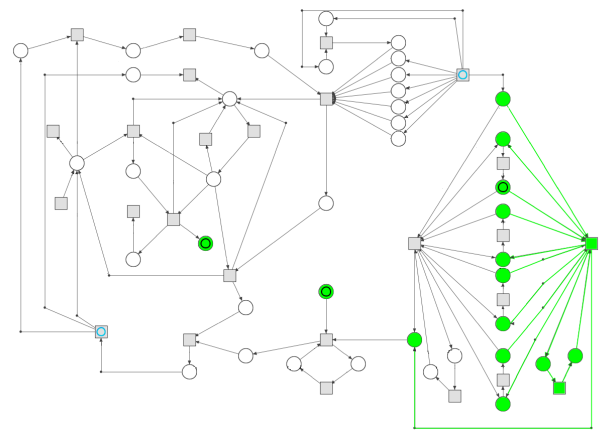
Rysunek 9.18: Podsieć GG-8



Rysunek 9.19: Podsieć GG-9



Rysunek 9.20: Podsieć TC-9

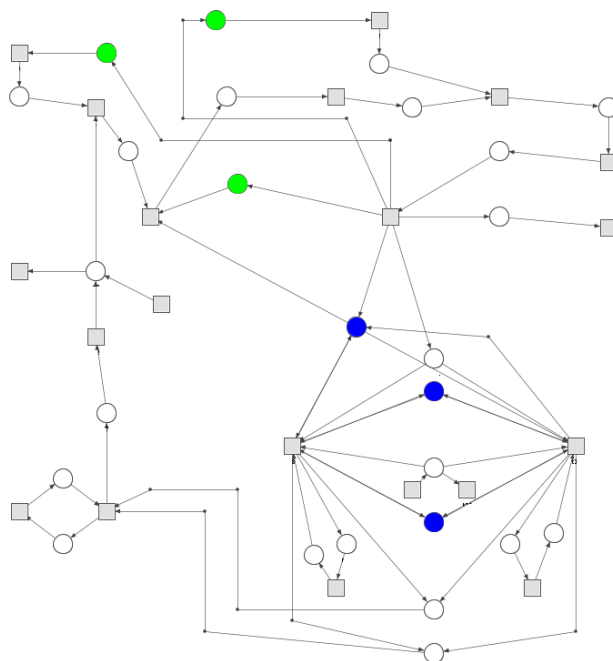


Rysunek 9.21: Podsieć TC-10

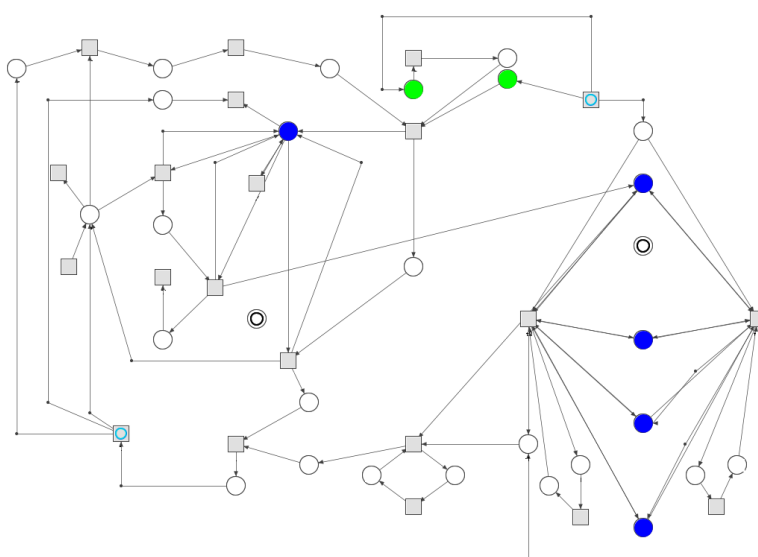
Na przykładzie podsieci GG_8 , GG_9 oraz TC_9 , TC_{10} (Rysunki 9.18 – 9.21), przedstawiono przypadki idealnych dopasowań. Opisują one izomorficzne podsieci o rozmiarze 14 wierzchołków, zbudowane wokół jednej tranzycji rozgałęziającej. W obu modelach GG-NER i TC-NER, podsieci GG_8 , GG_9 oraz TC_9 , TC_{10} sąsiadują ze sobą współdzieląc aż 10 miejsc granicznych. Różnice pojawiają się, gdy wiedza o podprocesach zostanie uzupełniona używając informacji o podsieciach trywialnych (przepływowych) z poprzedniej sekcji. Wskazują one, że funkcje zmapowanych miejsc granicznych nie zawsze są takie same, co na podstawie analizy samodzielnych struktur nie byłoby osiągalne do wykazania.

9.4.4 Porównanie sieci po redukcjach strukturalnych

W tym miejscu zostanie przedstawione przykładowe zastosowanie wspomnianych już redukcji strukturalnych sieci Petriego (opisanych w sekcji 2.2.3) jako narzędzia wspomagającego metodę porównywania zdekomponowanych podsieci.



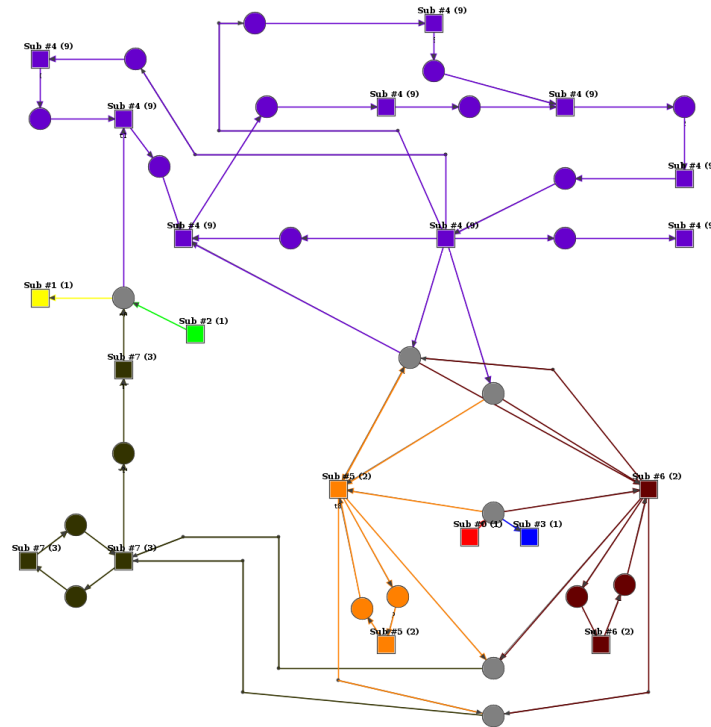
Rysunek 9.22: Model GG-NER po wstępnych redukcjach niezmieniających liczby występujących t-niezmenników



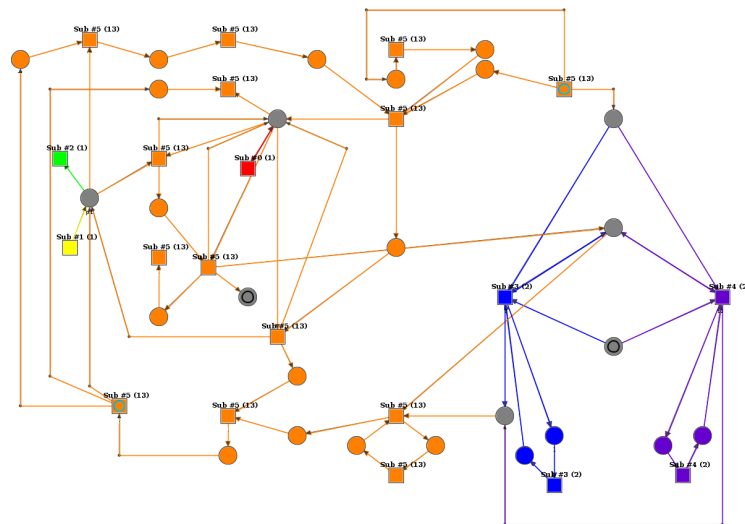
Rysunek 9.23: Model TC-NER po wstępnych redukcjach niezmieniających liczby występujących t-niezmenników

Przed przeprowadzeniem redukcji należy zaznaczyć, że modele z Rysunku 9.11 zostały stworzone w celu przeprowadzenia na nich analiz opartych o t-niezmenniki. Wpłynęło to na struktury obu modeli i pośrednio na potencjalne redukcje, jakie mogą zostać na nich przeprowadzone. W obu modelach GG-NER, oraz TC-NER są to redukcje strukturalne numer 1 i 2 z Rozdziału 6 [83]. Ich główną cechą jest

to, że nie zmieniają one liczby t/p-nieziemienników, pozostawiając tym samym model uproszczonym, ale niezmiennym z matematycznej perspektywy.



Rysunek 9.24: Sieć GG-NER po usunięciu łuków odczytu i połączonych z nimi miejsc w wyniku działania redukcji numer 6, oraz po wyznaczeniu nowych t-sieci.



Rysunek 9.25: Sieć TC-NER po usunięciu łuków odczytu i połączonych z nimi miejsc w wyniku działania redukcji numer 6, oraz po wyznaczeniu nowych t-sieci.

Natomiast użycie redukcji numer 3, która jest możliwa w obu modelach, wpłynie na licznosc zbioru minimalnych t-nieziemienników, podczas gdy liczba p-nieziemienników pozostanie bez zmian. Odwrotna sytuacja zachodzi dla redukcji numer 4. Nie wyklucza to ich użycia, a wymusza wychwycenie, które

niezmienniki zostały połączone w nowy w wyniku zajścia redukcji numer 3 lub 4. Jest to pożądana sytuacja w przypadku sieci o dużej liczbie niezmienników (ponad 100 tysięcy), gdzie ich redukcja upraszcza analizę.

W wyniku zastosowania redukcji numer 1 z modeli usunięto podsieci trywialne/przepływowe. Obszary podlegające redukcji zostały zaznaczone kolorem niebieskim na Rysunkach 9.22 i 9.23. Usunięcie tych podsieci nie wpłynęło na liczbę i rozmieszczenie t-komponentów (podsieci indukowanych przez t-niezmienniki), a zredukowało rozmiar (uproszcilo) części z nich i utworzyło luki odczytu, które podlegają osobnej redukcji w jednym z następujących kroków.

W wyniku zastosowania redukcji numer 4 równoległe ścieżki występujące pomiędzy dwoma tranzycjami rozgałęziającymi zostały zastąpione przez pojedynczą ścieżkę – oznaczoną zielonym kolorem na modelach.

Kolejnym krokiem jest zastosowanie rozszerzonej redukcji numer 6. Domyślnie usuwa ona miejsce połączone tylko za pomocą łuku odczytu z resztą sieci. W częściowo zredukowanych sieciach GG i TC występują miejsca połączone z kilkoma tranzycjami tylko za pomocą łuków odczytu. Oznacza to, że liczba tokenów w tych miejscach jest stała. Efekt redukcji jest przedstawiony na Rysunkach 9.24 i 9.25.

Tabela 9.4: Wyniki porównania podobieństwa zredukowanych modeli GG-NER i TC-NER

	TC_0^R	TC_1^R	TC_2^R	TC_3^R	TC_4^R	TC_5^R
GG_0^R	0/2	2/2	1/2	2/2	2/2	2/2
GG_1^R	0/2	1/2	2/2	2/2	2/2	2/2
GG_2^R	0/2	2/2	1/2	2/2	2/2	2/2
GG_3^R	0/2	1/2	2/2	2/2	2/2	2/2
GG_4^R	2/23	2/23	2/23	5/23	4/23	15/23
GG_5^R	3/9	2/9	2/9	5/9	6/9	5/9
GG_6^R	3/9	2/9	2/9	5/9	6/9	5/9
GG_7^R	3/9	2/9	2/9	5/9	6/9	5/9

Wnioski

Analizując przeprowadzone w tej sekcji redukcje sieci oraz ich wpływ na wynik porównania sieci zostały wysunięte następujące wnioski.

Użycie redukcji strukturalnej numer 1 pozwala na usunięcie z modelu trywialnych podsieci przepływowych. Efektem jest zmniejszenie rozmiarów tabeli wynikowej (Tabela 9.3 odpowiada wynikom porównania na sieci sprzed redukcji, a Tabela 9.4 po redukcjach opisanych w poprzedniej sekcji). Taki sam efekt może zostać uzyskany bez przeprowadzania redukcji sieci, a poprzez ograniczenie zbioru porównywanych podsieci przez pominięcie trywialnych podsieci. Utrata informacji, jaka musi zajść w efekcie takich działań, jest w przypadku badanych modeli na akceptowalnym poziomie. Należy jednak pamiętać, że dotyczy to każdego możliwego porównania.

Pozostałe z przeprowadzonych redukcji miały wpływ na rozmiar nietrywialnych podsieci uczestniczących w procesie porównania. Zwracając uwagę na redukcje przeprowadzone na największych podsieciach GG_7 i TC_{11} . Podlegały one redukcjom różnych typów, kolejno trzem redukcjom numer 3 na GG_7 i trzem redukcjom numer 3 i jednej redukcji numer 4. Mimo uproszczenia obu struktur największa wspólna podsieć wykazuje ten zbliżony stopień podobieństwa wynoszący około 63% całej struktury.

Inna sytuacja zachodzi dla czwórki podsieci GG_8 , GG_9 , TC_9 i TC_{10} , które w niemodyfikowanych

sieciach przedstawiały idealne dopasowania między sobą. Po redukcjach poziom spadł do 55 – 67%. Mimo zmniejszenia się znajdowanych wspólnych podsieci, wciąż reprezentują one najlepsze dopasowania między sobą. Pytaniem otwartym pozostaje, czy taka sytuacja jest pożądana. Odpowiedź na nie będzie zależna od kontekstu biologicznego porównanych sieci.

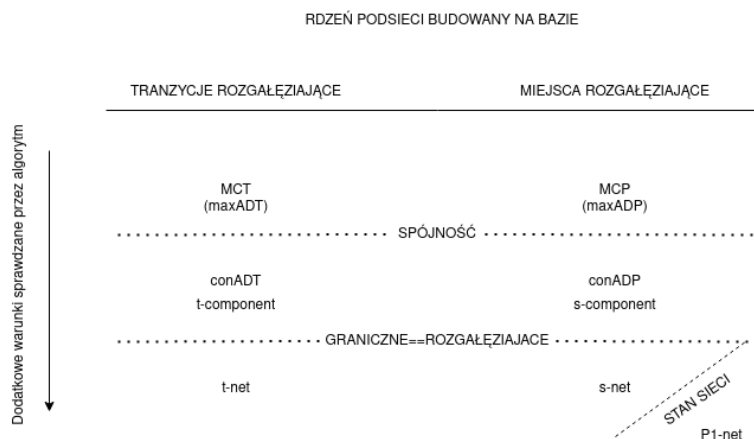
Porównując tabele wynikowe porównań podsieci dla modeli redukowanych, jak i oryginalnych, można zauważyć, że dopasowania dla podsieci o największych rozmiarach przetrwały proces uproszczenia. Przedstawia to sytuację, gdzie uproszczenie modeli zostało przeprowadzone na poziomie, który nie zaburzył relacji obserwowanych pomiędzy nieredukowanymi sieciami. Jednocześnie przypadek czterech podsieci GG_8 , GG_9 , TC_9 i TC_{10} odpowiada sytuacji, gdzie redukcje zmniejszyły rozmiar największych wspólnych podsieci, jakie w nich występują.

Wychodząc poza opisywany powyżej przypadek redukcji i porównania, możliwe jest postawienie dodatkowych wniosków.

Możliwość uproszczenia struktur modeli w celu uzyskania większej wspólnej podsieci (lepszego dopasowania wierzchołków) jest pożądana i pozwala na wykrycie faktycznego stopień podobieństwa. Jednocześnie należy po raz kolejny zaznaczyć, że redukcje przeprowadzone bez uprzedniego sprawdzenia ich wpływu na informację zawartą w modelu, mogą przynieść efekt odwrotny od zamierzonego. Z tego powodu konieczny jest nadzór użytkownika z wiedzą ekspercką, co pozwoli np. określić, czy pełne dopasowanie nietrywialnych struktur jest efektem przypadkowym, czy redukcja strukturalna jest konieczna, a jeśli tak, to na którym poziomie wprowadzanych uproszczeń proces powinien zostać zatrzymany, aby uzyskać najlepsze rezultaty.

9.5 Zastosowanie do pokrewnych dekompozycji opartych o niezmienniki

Ze względu na współdzielenie wielu własności przez dekompozycje oparte o niezmienniki które zostały przeanalizowane w Rozdziałach 4 i 5, opisany algorytm przy niewielkich zmianach może zostać użyty do porównania innych podsieci należących do tej rodziny dekompozycji bazujących na niezmiennikach. Punktem wyjściowym są podsieci t-sieci i opisywane różnice oraz modyfikacje w algorytmie będą przedstawione z ich perspektywy. Relacje i wpływ poszczególnych dekompozycji na algorytm porównywania zostały przedstawione na Rysunku 9.26. Szczegóły dotyczące konkretnych dekompozycji opisano w dalszej części rozdziału.



Rysunek 9.26: Relacje pomiędzy algorytmem porównywania a dekompozycjami

9.5.1 conADT

Główną różnicą pomiędzy strukturami indukowanymi przez zbiory conADT a t-sieci jest to, że dla tych pierwszych zbiorów wierzchołków rozgałęziających nie zawsze będzie pokrywał się ze zbiorem miejsc granicznych – przypadek z Rysunku 9.2 gdzie zbiór miejsc rozgałęziających może być liczniejszy.

Modyfikacje algorytmu: Ze względu na to, że w sieciach indukowanych przez conADT zbiór miejsc rozgałęziających nie musi ograniczać struktury budowanej od tranzycji rozgałęziających, konieczne było dodanie warunku sprawdzającego, czy z miejsca granicznego wychodzi ścieżka P-P lub P-T z tranzycją końcową o stopniu wierzchołka równym jeden, należąca do podsieci i w przypadku znalezienia takiej, kontynuowania porównywania elementów tej ścieżki. Sytuacja ta zachodzi tylko dla wspomnianych dwóch przypadków ścieżek. W pozostałych będą udział brać tranzycje rozgałęziające i rozbudowując od nich wspólną podsieć pokryta/przeszukana zostanie cała podsieć.

9.5.2 MCT

Podsieci indukowane przez zbiory MCT składają się z przynajmniej jednego komponentu w postaci podsieci indukowanej zbiorem conADT. Z tego powodu współdzielą z nimi wszystkie charakterystyki z wyłączeniem spójności podsieci (struktura w ramach komponentu conADT musi pozostawać spójna).

Modyfikacje algorytmu: Aby dostosować algorytm do operowania na omawianym typie podsieci, należało usunąć warunek spójności, omawiany w sekcji 9.3.3, który jest domyślnie włączony dla t-sieci i conADT, oraz zastosować zmiany opisane dla conADT.

9.5.3 T/S-komponent

Struktury indukowane przez t- oraz s-komponenty są przez algorytm traktowane jak podsieci bazujące na zbiorach conADT.

Modyfikacje algorytmu: Istotną różnicą jest liczba wierzchołków rozgałęziających występujących w pierwszej dekompozycji, która wpływa negatywnie na czas porównania. Zaproponowana heurystyka ograniczająca wstępnie przestrzeń dopasowań nie jest wystarczająco efektywna, aby operować na strukturalnie bardziej skomplikowanych t/s-komponentach.

9.5.4 ADP/MCP – podsieci bazujące na p-niezmiennikach

Algorytm jest kompatybilny z dekompozycjami do podsieci indukowanych przez zbiory ADP. Koniecznym jest odwrócenie relacji pomiędzy tranzycjami a miejscami rozgałęziającymi. Jest to spowodowane tym, że te dekompozycje są tworzone na bazie p-niezmienników, a nie t-niezmienników.

Modyfikacje algorytmu: Zmiana danych wejściowych i budowanie wspólnych podsieci traktując miejsca rozgałęziające jako rdzeń struktury.

P1 sieć/SMC

Elementem wyróżniającym podsieci P1 od podsieci indukowanych zbiorem ADP jest uwzględnianie stanu i ograniczenie do jednego tokenu w całej podsieci. Potencjalnym rozszerzeniem algorytmu porównywania jest uwzględnienie lokalizacji tokenu przy określaniu podobieństwa podsieci.

9.5.5 s-sieci

W przypadku podsieci s-sieci, które nie są oparte o niezmienniki, ale podobnie jak siostrzane t-sieci są zbudowane na bazie indeksowanych miejsc, na potrzeby tego algorytmu są one traktowane tak samo jak sieci P1 i SMC.

9.6 Porównanie przez dekompozycję – podsieci funkcyjne

W przypadku sieci opartych o t-niezmienniki, t-sieci oraz sieci funkcyjnych tranzycja jest głównym, elementem wokół którego budowane są te podsieci. Mimo takiego podobieństwa pomiędzy tymi typami podsieci algorytm zaproponowany w poprzedniej części dla MCT i t-sieci nie nadaje się do użycia przy porównaniu sieci funkcyjnych. Głównym powodem jest niesekwencyjność sieci funkcyjnych (żadne dwie tranzycje należące do jednej sieci nie mogą posiadać miejsca, które dla jednej z nich jest miejscem wejściowym, a dla drugiej miejscem wyjściowym). Odpowiadają one podsieciom w których wszystkie tranzycje mogą zostać uruchomione jednocześnie (równolegle). Natomiast podsieci oparte o t-niezmienniki głównie reprezentują sekwencyjne fragmenty.

Rezultatem wspomnianych charakterystyk sieci są istotne różnice strukturalne, które mają postać całkowitego braku miejsc przepływowych w ramach podsieci funkcyjnej (miejsce przepływowe może istnieć w dekomponowanej sieci, natomiast nie będzie występować w tej postaci w żadnej z podsieci, a jako miejsce wyjściowe w jednej z podsieci i miejsce wejściowe w drugiej). Jednocześnie zbiór miejsc granicznych nie zawiera tylko i wyłącznie miejsc rozgałęziających, ale także wejściowe i wyjściowe. Wierzchołki rozgałęziające bez względu na typ będą powiązane z nietrywialnymi podsieciami, co jest kolejną różnicą w porównaniu z podsieciami bazującymi na konkretnym typie niezmiennika.

Tranzycje rozgałęziające nie są jedynym nośnikiem informacji i współdzielą tą rolę z miejscami rozgałęziającymi. W rezultacie zbiór potencjalnych sieci funkcyjnych tylko w małym fragmencie pokrywa się ze strukturami bazującymi na t-niezmiennikach. Analizując nietrywialne podsieci funkcyjne, można je podzielić na trzy zbiory. Należy też zaznaczyć istnienie trywialnych podsieci funkcyjnych. Reprezentują je struktury niezawierające w sobie żadnych wierzchołków rozgałęziających. Różnice pomiędzy podsieciami funkcyjnymi a innymi dekompozycjami opartymi o t-niezmienniki określa relacja pomiędzy tranzycjami a miejscami rozgałęziającymi. Dla podsieci opartych o t-niezmienniki oznaczają one w większości przypadków ich granicę, podczas gdy dla podsieci funkcyjnych są elementem łączącym równolegle tranzycje w większą podsieć.

Pierwszą grupę reprezentują podsieci zbudowane wokół pojedynczej tranzycji rozgałęziającej, bez miejsc rozgałęziających. Efektem jest struktura drzewiasta w której najdłuższa możliwa ścieżka będzie miała długość 2 łuków. Podsieci z tej grupy mogą być porównywane z użyciem algorytmu dla podsieci bazujących na niezmiennikach.

Drugą grupę stanowią podsieci zawierające miejsca rozgałęziające i tylko tranzycje przepływowe. Takie struktury mogą zostać porównane opisanym wcześniej algorytmem z tą różnicą, że podsieci są traktowane jak zbudowane na bazie p-niezmienników. Takie podejście jednak nie pozwala na porównanie z podsieciami należącymi do pierwszej grupy. Możliwość porównania do podsieci zawierających tylko miejsce rozgałęziające jest znaczącym ograniczeniem stanowiącym przesłankę do opracowania dedykowanego algorytmu.

Trzecia grupa zawiera podsieci zawierające w sobie jednocześnie oba typy wierzchołków rozgałęziających, a przez to o największym potencjale pod względem rozmiaru zawartych w nich informacji. Nie są one kompatybilne z algorytmem przewidzianym dla sieci opartych o niezmienniki.

Głównym powodem niekompatybilności jest to, że algorytm buduje wspólną podsieć na podstawie wierzchołków rozgałęziających jednego typu (tranzycji). Następnie uzyskany rdzeń jest poszerzany o wspólne ścieżki zakończone rozgałęziającym wierzchołkiem drugiego typu (miejsce). W przypadku podsieci funkcyjnych mechanizm ten działa w ograniczonej postaci dla podsieciach należących do pierwszej grupy – brak sekwencyjności w sieciach funkcyjnych ogranicza go do jednej tranzycji rozgałęziającej która nie może być połączona miejscem przepływowym z inną tranzycją rozgałęziającą w ramach jednej podsieci. Algorytm musiałby zostać przebudowany do mapowania w pierwszym kroku ścieżek występujących pomiędzy wierzchołkami rozgałęziającymi niezależnie od ich rodzaju, a nie tak jak został stworzony do mapowania ścieżek pomiędzy wierzchołkami rozgałęzionymi tego samego typu. Taka zmiana implikuje liczne dodatkowe modyfikacje. Z tego powodu po wstępnych próbach zarzucony został pomysł modyfikacji wcześniej opracowanej metody na rzecz opracowania nowego algorytmu bazującego na charakterystykach sieci funkcyjnych.

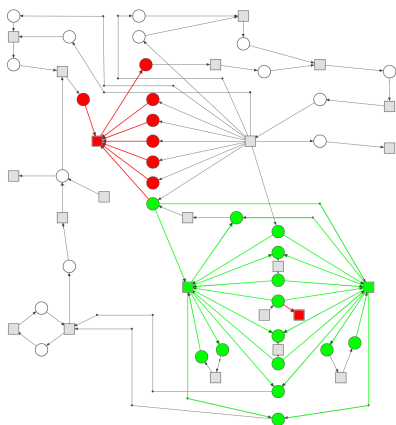
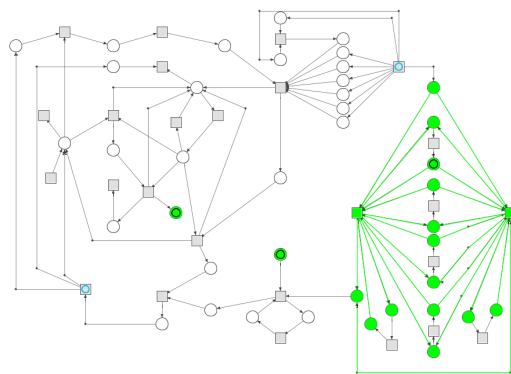
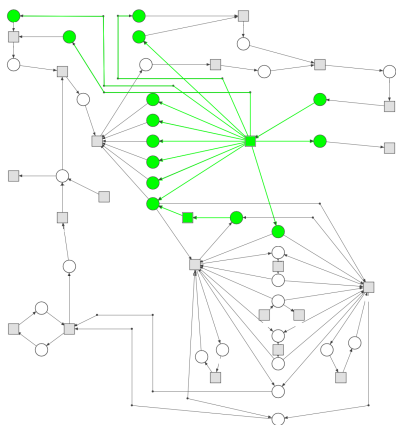
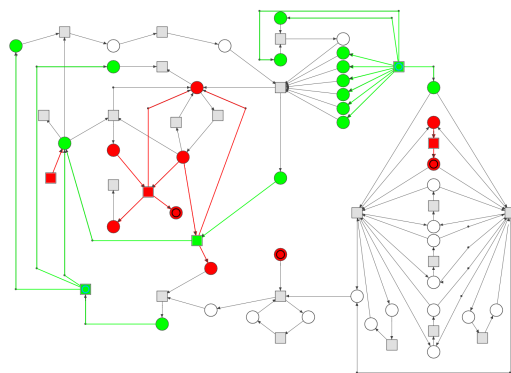
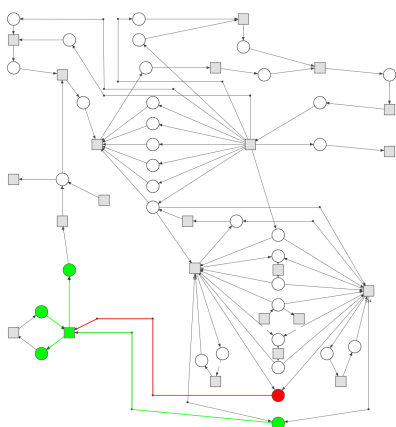
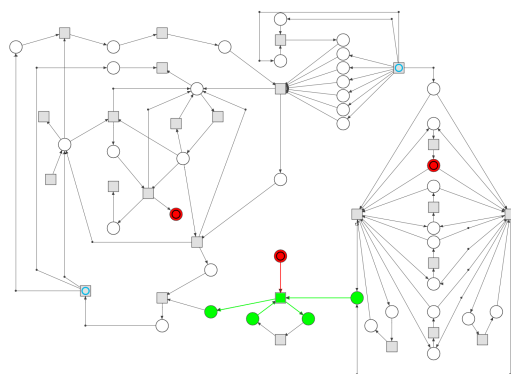
9.6.1 Podsieć

Podobnie jak w przypadku wersji dla sieci bazujących na niezmiennikach, przed rozpoczęciem pracy nad algorytmem zostało postawione pytanie odnośnie postaci wspólnego podgrafu. Ze względu na funkcje i relacje, jakie zachodzą pomiędzy typami wierzchołków w sieciach funkcyjnych, rozważane były trzy warianty budowy wspólnej podsieci:

- brak preferencji typu wierzchołków przy budowie podsieci (największa wspólna podsieć),
- maksymalizacja miejsc występujących w podsieci kosztem tranzycji,
- maksymalizacja tranzycji występujących w podsieci kosztem miejsc.

Powyższe warianty zostały omówione w rozmowie z autorem sieci funkcyjnych. Wnioski wyniesione z niej wskazują, że każdy z proponowanych wariantów ma podstawy do reprezentacji wartościowej informacji wewnątrz modelu. Jednak to od kontekstu w jakim zostały utworzone porównywane modele zależy czy i jakie znaczenie ma macierz incydencji odpowiadająca sieci Petriego, czy w naszym przypadku jej fragmentowi.

Jako że sieci funkcyjne nie są popularne w zastosowaniach biologicznych, po konsultacji z zespołem jako najbardziej odpowiadający dla budowanych modeli został wybrany wariant preferujący tranzycje. W zależności od wyników przyszłych prac i analiz z użyciem sieci funkcyjnych, pozostałe warianty mogą zostać rozważone ponownie.

Rysunek 9.27: Wspólna podsieć dla GG_2 i TC_5 Rysunek 9.28: Wspólna podsieć dla TC_5 i GG_2 Rysunek 9.29: Wspólna podsieć dla GG_7 i TC_3 Rysunek 9.30: Wspólna podsieć dla TC_3 i GG_7 Rysunek 9.31: Wspólna podsieć dla GG_{13} i TC_{12} Rysunek 9.32: Wspólna podsieć dla TC_{12} i GG_{13}

9.7 Podsumowanie

Dużą zaletą opisanej metody jest możliwość operowania na całej rodzinie podsieci wyznaczanych na podstawie niezmienników. Zwiększa to potencjał metody przez możliwość operowania na dekompozycjach stosowanych w różnych obszarach naukowych, nie tylko dla systemów biologicznych. Relacje uszczegółowienia obserwowane pomiędzy pewnymi dekompozycjami (np. między podsieciami indukowanymi przez MCT, conADT i t-sieciami) pozwalają na wybór typu podsieci lepiej opisującego poszukiwane podprocesy.

Jednocześnie największego potencjalnego ograniczenia powiązanego z wykładniczym wzrost liczby cząstkowych wektorów przy wyznaczaniu niezmienników można uniknąć przez zastąpienie podsieci bazujących na niezmiennikach podsieciami t-sieci lub s-sieci. Mimo że reprezentują mniejsze struktury i nie są bezpośrednio powiązane z niezmiennikami, pozwalają na przeprowadzenie części analiz z zachowaniem części informacji oczekiwanych dla podsieci indukowanych przez zbiory conADT i MCT.

Kolejną zaletą jest możliwość użycia jednego z kilku wariantów wspólnej podsieci zwracanej przez algorytm. Pozwalają one na lepsze dopasowanie poszukiwanego rezultatu do kontekstu, w którym szukana jest wspólna podsieć, zwłaszcza w przypadkach porównania sieci pochodzących z różnych źródeł, przez możliwość dostosowania wspólnej podsieci do pożądanej postaci.

Jak każda z omawianych metod posiada ona pewne ograniczenia, których użytkownik musi być świadom, aby w pełni wykorzystać jej możliwości.

Metoda zwraca informacje na temat stopnia strukturalnego podobieństwa pomiędzy podsieciami. Całkowicie pomijana jest kwestia dopasowania etykiet w znalezionych podsieciach, choć rozszerzenie algorytmu o możliwość sprawdzenia etykiet podczas wyznaczania wspólnej podsieci jest możliwe i proste implementacyjnie. Problem dopasowania etykiet w sieciach biologicznych został szerzej opisany w Rozdziale 7.

Rozdział 10

Wyniki uzupełniających eksperymentów obliczeniowych

10.1 Wprowadzenie

Metody porównywania opisane w tej pracy reprezentują różne podejścia do określania stopnia podobieństwa pomiędzy parą modeli układów biologicznych. Różnice w sposobie działania, różne definicje istotności elementu sieci powodują, że w niektórych przypadkach bezpośrednie porównanie wyników zwróconych przez rozpatrywane metody może prowadzić do błędnych wniosków. Źródła zaistniałej sytuacji leżą u podstaw tych metod. Reprezentują one różne podejścia do problemu porównywania sieci Petriego. Metoda z Rozdziału 7 dokonuje porównania sieci na bazie zachodzących w nich przepływów przedstawionych za pomocą t-niezmienników, podczas gdy metody z pozostałych rozdziałów badają podobieństwo strukturalne. Metoda bazująca na dekompozycji z Rozdziału 9 jest w stanie wykazać izomorfizm sieci, podczas gdy metody bazujące na grafietych i wierzchołkach rozgałęziających z Rozdziałów 6 i 8 nie są w stanie tego wykazać (np. identyczność rozkładów pn-grafiletów jest jedynie przesłanką). Z tych powodów konieczna jest znajomość możliwości, a zwłaszcza ograniczeń każdej z opisywanych metod. Pozwoli to na właściwy dobór podejścia dla porównywanych sieci i uzyskanie wyniku, który zostanie prawidłowo zinterpretowany.

W tym celu do przykładów porównywania znanych z Rozdziałów 6-9 przeprowadzono uzupełniające porównania, których wyniki zostały zaprezentowane w niniejszym rozdziale.

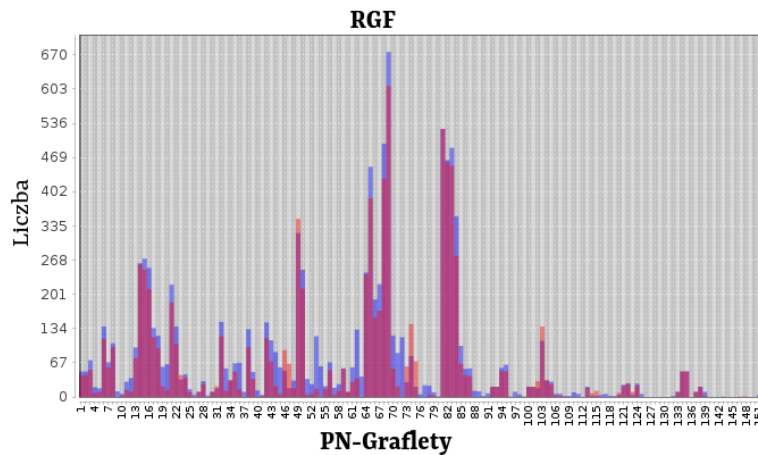
10.2 Uzupełnienie przypadku – modele naprawy DNA

10.2.1 Grafilety

Relative Graphlet Frequency distance – RGF

Porównanie z użyciem metryki RGF sieci GG-NER i TC-NER zwróciło wartość 300,645 (rozkład poszczególnych pn-grafiletów przedstawiono na Rysunku 10.1). Zestawiając dystans mierzony przez metrykę z wynikami uzyskanymi dla porównań małych sieci metabolicznych SM I z SM II oraz SM II z SM III z Rozdziału 8 przedstawionych w Tabeli 8.2 można stwierdzić że reprezentują one podobny poziom. Opisywane tam przypadki były określone jako modele reprezentujące niepowiązane ze sobą procesy. Jednak błędnym wnioskiem byłoby uznanie GG-NER i TC-NER za nie powiązane ze sobą

procesy na bazie tego, że reprezentują sieci o rozmiarach podobnych do małych sieci metabolicznych o SM I, SM II i SM III, oraz mają podobnej wielkości wartość zwracaną przez metrykę RGF. RGF skupia się na wielkości dystansu pomiędzy występowaniem danego grafletu w obu porównywanych sieciach. Liczba wystąpień wszystkich grafletów w sieciach z podrozdziału 8.5.1 wynosi około 2000, podczas gdy w rozpatrywanym przypadku mowa o około 8000. W efekcie obserwowana jest inna skala różnic: wystąpienia pn-grafletów z porównania sieci SM I z SM II sięgały 90 w najliczniejszych przypadkach, podczas gdy dla GG i TC mowa o 609 i 676 znalezionych pn-grafletach G_{68}^{PN} , gdzie sam dystans wynosi 67. Zagęszczenie występujących pn-grafletów (nie mylić z gęstością sieci) ma wpływ na interpretację wyniku zwróconego RGF.



Rysunek 10.1: Rozkład pn-grafletów pomiędzy sieciami GG-NER a TC-NER przedstawionymi na Rysunku 9.11.

GDDA

Porównanie z użyciem metryki GDDA zwróciło wartość 1,2445 (to że wynik GDDA może przyjąć wartość powyżej 1 opisano w erracie [96] do pracy [98]), co wskazuje na wysokie podobieństwo strukturalne porównywanych modeli GG-NER i TC-NER. Zaznaczmy, że dla obu przypadków gęstość sieci wynosi 0,105, co pozwala określić wynik porównania z użyciem GDDA jako stabilny. Jednocześnie rozmiar macierzy GDD nie pozwala na wygodne analizowanie wyników cząstkowych, jak to miało miejsce dla porównania z użyciem RGF.

10.2.2 Relative Branching Vertices Frequency distance – RBF

Wynik uzyskany za pomocą metryki RBF dla porównania sieci NER wynosi 33,9 (kolejno 13,28 i 14,04 dla tranzycji i miejsc). W porównaniu z pozostałymi przykładami wskazuje to na stosunkowo małą liczbę różnic dla sieci o rozpatrywanym rozmiarze. Przechodząc do danych cząstkowych (Tabela 10.1 i Rysunki 10.2) zaobserwowana jest podobny rozkład wierzchołków rozgałęziających dla około 60% przypadków. Jest to też obserwowane osobno dla tranzycji i miejsc rozgałęziających. Co istotne dla tego porównania, takie same wierzchołki rozgałęziające najwyższych stopni są obecne w obu sieciach. Jest to przesłanka do istnienia większej (rozbudowanej) wspólnej struktury w sieciach. Jednocześnie liczba wierzchołków rozgałęziających występujących tylko w jednej sieci, wraz z dystansem RBF, wskazują na istnienie znaczących różnic strukturalnych.

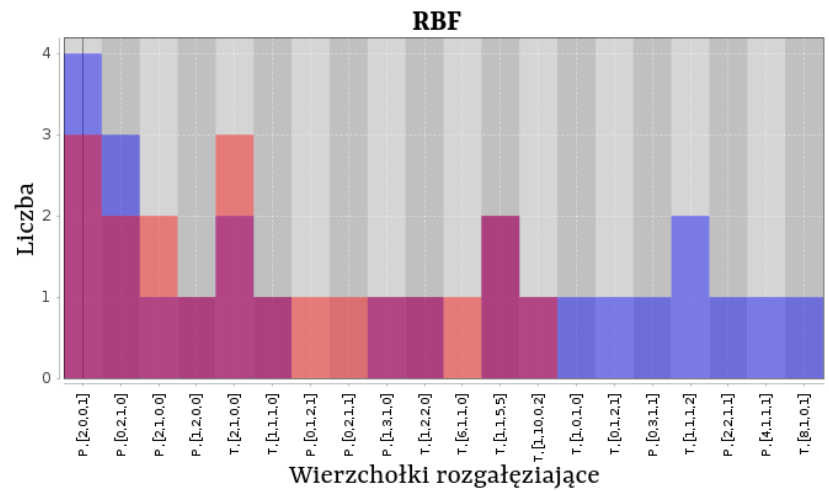
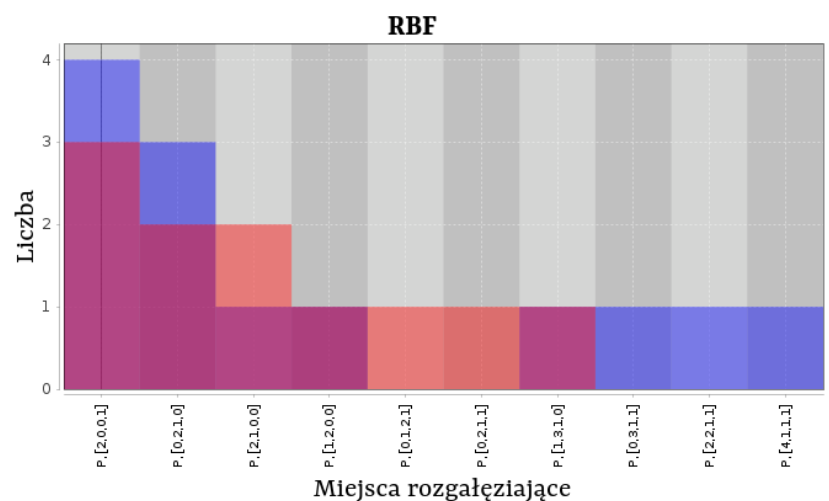
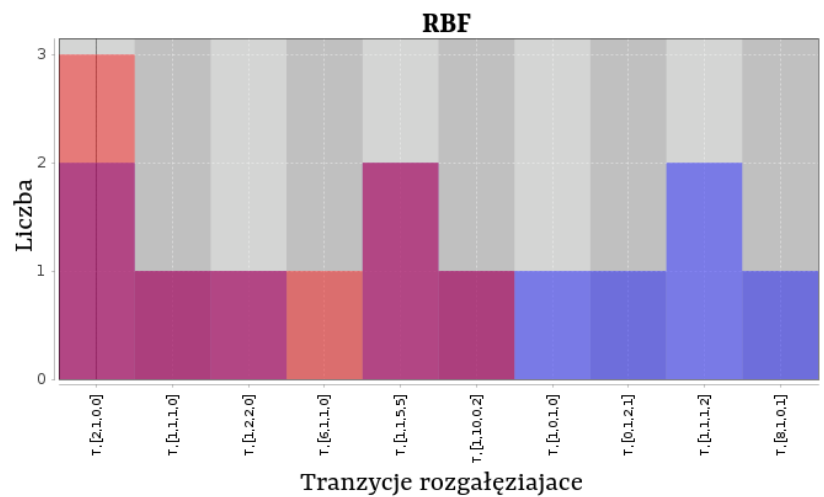


Tabela 10.1: Wyniki cząstkowe dla RBF

Wierzchołek Rozgałęziający	WCząstkowy RBF
T [2,1,0,0]	0,629
T [1,1,1,0]	0,223
T [8,1,0,1]	3,219
T [1,2,2,0]	0,223
T [6,1,1,0]	2,996
T [1,1,5,5]	0,223
T [1,10,0,2]	0,223
T [1,0,1,0]	3,219
T [0,1,2,1]	3,219
T [1,1,1,2]	2,526
P [0,3,1,1]	3,219
P [2,2,1,1]	3,219
P [4,1,1,1]	3,219
P [2,0,0,1]	0,065
P [0,2,1,0]	0,182
P [2,1,0,0]	0,916
P [1,2,0,0]	0,223
P [0,1,2,1]	2,996
P [0,2,1,1]	2,996
P [1,3,1,0]	0,223



Rysunek 10.2: Trzy diagramy prezentujące rozkład wierzchołków rozgałęziających dla sieci GG-NER i TC-NER.

10.3 Uzupełnienie przypadku – sieci chorobowe

10.3.1 Grafilety – GDDA

W ramach uzupełnienia porównań trzech modeli chorobowych MCh I, MCh II, MCh III z sekcji 8.5.2 przeprowadzonych z użyciem metryki RGF i RBF, dokonano porównania z użyciem metryki GDDA. Wyniki porównań tych modeli przedstawiono w Tabeli 10.2

Tabela 10.2: Wyniki porównań pomiędzy parami sieci chorobowych MCh I, II i III uzyskane z zastosowaniem metryki GDDA.

Sieć I	Sieć II	GDDA
MCh I	MCh II	0,051
MCh I	MCh III	0,050
MCh II	MCh III	0,611

Porównując je z rezultatami RGF (opisane w sekcji 8.5.2) widoczne jest, że obie metryki określają sieci chorobowe o numerach 5 i 6 jako podobne, od których znacząco odróżnia się wariant sieci o numerze 4. Różnica jest w skali podobieństwa, zwłaszcza z perspektywy GDDA, której wartość wynosząca 0,611 wskazuje na relatywnie niski poziom podobieństwa.

Dodatkowo przeprowadzono test mający na celu ustalić skalę zmian GDDA w przypadku wyłączenia jednej tranzycji. W rezultacie powstały trzy podsieci modelu 3, z których każdy miał wyłączony jeden wierzchołek. Są to kolejno tranzycje t_5 , t_{42} i t_{59} . W wyniku porównania tych podsieci z siecią 3, uzyskano wyniki przedstawione w Tabeli 10.3.

Tabela 10.3: Wyniki porównań z zastosowaniem metryki GDDA pomiędzy siecią chorobową SM III a jej podsieciami z wyłączonej tranzycją t_5 , t_{42} lub t_{59} .

Sieć I	Sieć II (z wyłączonej tranzycją)	GDDA
MCh III	MCh III \ { t_5 }	0,079
MCh III	MCh III \ { t_{42} }	0,240
MCh III	MCh III \ { t_{59} }	0,895

Nawet takie niewielkie różnice w istotny sposób wpłynęły na wartość GDDA. Najmniejszy wpływ miało usunięcie tranzycji źródłowej t_{59} . Powstaje pytanie o powód takiej wrażliwości na różnice, jakie obserwujemy w danym przypadku. Odpowiedzią jest liczba grafiletów i w szczególności orbit znajdujących w poszczególnych wierzchołkach. W Tabelach 10.6, 10.4 i 10.5 przedstawione zostały wystąpienia orbit w usuniętych wierzchołkach. Liczba i zróżnicowanie orbit odpowiadają różnym sąsiedztwom wyłączonych tranzycji. Sytuacja taka jest obserwowana pośród modeli o wysokim zagęszczeniu grafiletów.

Tabela 10.5: Liczba poszczególnych orbit rozpatrywanych w ramach tranzycji t_{59} Tabela 10.4: Liczba poszczególnych orbit rozpatrywanych w ramach tranzycji t_{42}

	t_{42}		t_{42}
Orbita-0	2	Orbita-138	2
Orbita-3	1	Orbita-145	3
Orbita-4	3	Orbita-148	2
Orbita-6	5	Orbita-153	9
Orbita-7	3	Orbita-159	6
Orbita-9	5	Orbita-164	2
Orbita-12	2	Orbita-169	2
Orbita-17	1	Orbita-177	20
Orbita-18	3	Orbita-179	2
Orbita-20	10	Orbita-181	6
Orbita-22	1	Orbita-182	14
Orbita-24	25	Orbita-184	1
Orbita-25	9	Orbita-186	6
Orbita-27	6	Orbita-188	14
Orbita-33	1	Orbita-190	3
Orbita-38	2	Orbita-192	41
Orbita-40	9	Orbita-194	1
Orbita-42	2	Orbita-195	15
Orbita-48	3	Orbita-197	16
Orbita-50	3	Orbita-199	12
Orbita-52	2	Orbita-208	25
Orbita-56	3	Orbita-212	20
Orbita-58	4	Orbita-214	1
Orbita-60	2	Orbita-218	6
Orbita-62	5	Orbita-223	3
Orbita-66	3	Orbita-228	4
Orbita-68	9	Orbita-230	2
Orbita-70	1	Orbita-235	1
Orbita-78	1	Orbita-243	2
Orbita-90	4	Orbita-256	2
Orbita-92	9	Orbita-265	4
Orbita-94	10	Orbita-271	3
Orbita-95	3	Orbita-276	3
Orbita-97	8	Orbita-279	1
Orbita-99	9	Orbita-281	6
Orbita-100	9	Orbita-282	7
Orbita-104	13	Orbita-284	3
Orbita-105	5	Orbita-285	45
Orbita-110	3	Orbita-288	50
Orbita-118	2	Orbita-290	10
Orbita-119	4	Orbita-309	1
Orbita-121	9	Orbita-342	1
Orbita-123	4	Orbita-349	1
Orbita-124	11	Orbita-397	4
Orbita-126	8	Orbita-399	2
Orbita-129	10	Orbita-402	4
Orbita-133	2	Orbita-404	1
Orbita-135	8		
Σ		646	

	t_{59}		t_{59}
Orbita-0	1	Orbita-175	16
Orbita-3	2	Orbita-177	94
Orbita-4	5	Orbita-182	60
Orbita-6	9	Orbita-188	14
Orbita-9	8	Orbita-192	36
Orbita-12	2	Orbita-194	18
Orbita-15	1	Orbita-195	36
Orbita-20	16	Orbita-197	11
Orbita-22	10	Orbita-199	26
Orbita-24	42	Orbita-201	3
Orbita-25	14	Orbita-205	17
Orbita-36	1	Orbita-207	8
Orbita-38	5	Orbita-208	54
Orbita-40	9	Orbita-210	10
Orbita-42	2	Orbita-212	38
Orbita-44	5	Orbita-216	90
Orbita-54	24	Orbita-221	46
Orbita-58	19	Orbita-230	2
Orbita-60	9	Orbita-232	5
Orbita-62	13	Orbita-239	4
Orbita-64	6	Orbita-241	14
Orbita-66	7	Orbita-245	56
Orbita-68	7	Orbita-250	20
Orbita-72	1	Orbita-269	5
Orbita-76	1	Orbita-271	11
Orbita-89	3	Orbita-276	10
Orbita-90	13	Orbita-282	10
Orbita-92	39	Orbita-285	76
Orbita-94	24	Orbita-287	10
Orbita-95	6	Orbita-288	93
Orbita-100	3	Orbita-290	20
Orbita-102	13	Orbita-314	1
Orbita-104	29	Orbita-324	1
Orbita-105	5	Orbita-329	1
Orbita-107	6	Orbita-334	3
Orbita-112	6	Orbita-338	1
Orbita-116	62	Orbita-340	1
Orbita-119	7	Orbita-382	9
Orbita-121	30	Orbita-386	12
Orbita-123	12	Orbita-390	3
Orbita-128	8	Orbita-433	4
Orbita-129	35	Orbita-435	15
Orbita-131	5	Orbita-440	18
Orbita-133	9	Orbita-443	6
Orbita-135	19	Orbita-445	5
Orbita-138	3	Orbita-448	1
Orbita-140	7	Orbita-450	6
Orbita-151	5	Orbita-455	4
Orbita-153	12	Orbita-465	6
Orbita-156	7	Orbita-467	1
Orbita-159	6	Orbita-469	5
Orbita-162	24	Orbita-475	1
Orbita-167	8	Orbita-576	3
Σ		1645	

Tabela 10.6: Liczba poszczególnych orbit rozpatrywanych w ramach tranzycji t_5

	t_5
Orbita-0	1
Orbita-4	4
Orbita-22	5
Orbita-38	4
Orbita-42	5
Orbita-90	24
Orbita-95	2
Orbita-100	13
Orbita-105	12
Orbita-194	10
Orbita-207	13
Orbita-230	4
Orbita-235	1
Orbita-287	2
Σ	100

10.3.2 Porównania podsieci t-sieci

Używając metod opartych o dekompozycje dla niniejszych przypadków porównań natrafiamy na problem wykładniczego wzrostu liczby cząstkowych wektorów. Ogranicza to analizę do podsieci t-sieci, które nie są oparte o niezmienniki. Użycie tego typu dekompozycji dla rozpatrywanych modeli MCh I i MCh II owocuje dużą liczbą podsieci (66 dla pierwszej sieci i 59 dla drugiej), z których większość reprezentuje trywialne przypadki.

Tabela 10.7: Tabela wynikowa porównanych podsieci MCh I z przedziałów 55 – 65 i podsieci MCh II z przedziałów 50 – 58

	$MChII_{50}$	$MChII_{51}$	$MChII_{52}$	$MChII_{53}$	$MChII_{54}$	$MChII_{55}$	$MChII_{56}$	$MChII_{57}$	$MChII_{58}$
$MChI_{55}$	6/6	3/6	0/6	2/6	0/6	4/6	3/6	4/6	4/6
$MChI_{56}$	3/9	9/9	0/9	4/9	0/9	5/9	5/9	5/9	7/9
$MChI_{57}$	0/7	0/7	7/7	0/7	5/7	0/7	0/7	0/7	0/7
$MChI_{58}$	2/6	4/6	0/6	6/6	0/6	2/6	4/6	2/6	6/6
$MChI_{59}$	0/5	0/5	5/5	0/5	5/5	0/5	0/5	0/5	0/5
$MChI_{60}$	4/10	5/10	0/10	2/10	0/10	10/10	3/10	6/10	6/10
$MChI_{61}$	4/8	5/8	0/8	4/8	0/8	4/8	5/8	4/8	7/8
$MChI_{62}$	3/5	5/5	0/5	4/5	0/5	3/5	5/5	3/5	5/5
$MChI_{63}$	4/6	5/6	0/6	2/6	0/6	6/6	3/6	6/6	6/6
$MChI_{64}$	4/10	7/10	0/10	4/10	0/10	8/10	5/10	6/10	8/10
$MChI_{65}$	4/20	7/20	0/20	6/20	0/20	6/20	5/20	6/20	20/20

Przeanalizowano część podsieci reprezentujących nietrywialne struktury, których wyniki przedstawiono w Tabeli 10.7. Zaobserwowano pełne dopasowanie dla 9 przypadków, w tym dla największych podsieci ($MChI_{65}$ i $MChII_{58}$) o rozmiarze 20 wierzchołków. Interesujące są dwa przypadki podsieci z modelu MCh I dla których nie znaleziono pełnego dopasowania. Są to podsieci $MChI_{61}$ i $MChI_{64}$. Pierwsza z nich w wyniku zmiany jednego z miejsc przepływowych na miejsce rozgałęziające została podzielona na dwie trywialne podsieci t-sieci o $MChII_{15}$ i $MChII_2$. Natomiast druga podsieć reprezentowała wyłączony podproces chorobowy i nie istnieje w drugim modelu.

10.4 Małe sieci metaboliczne

10.4.1 Porównywanie z użyciem dekompozycji

Do porównania trzech małych sieci metabolicznych SM I, SM II i SM III, użyte zostały podejścia dekomponujące je do podsieci t-sieci (Tabela 10.9) i conADT (Tabela 10.8) pomiędzy którymi wyznaczono wspólne podsieci.

Tabela 10.8: Wynik porównań podsieci **conADT** występujących w sieciach *SMI* z Rysunku 8.4 i *SMII* z Rysunku 8.5

	$SMII_0^A$	$SMII_1^A$	$SMII_2^A$	$SMII_3^A$	$SMII_4^A$	$SMII_5^A$	$SMII_6^A$
SMI_0^A	5/20	0/20	4/20	2/20	9/20	2/20	7/20
SMI_1^A	0/3	0/3	0/3	0/3	0/3	0/3	0/3
SMI_2^A	0/3	0/3	0/3	0/3	0/3	0/3	0/3
SMI_3^A	7/31	0/31	4/31	2/31	10/31	2/31	7/31
SMI_4^A	0/3	0/3	0/3	0/3	0/3	0/3	0/3

Tabela 10.9: Wynik porównań podsieci **t-sieci** występujących w sieciach *SMI* z Rysunku 8.4 i *SMII* z Rysunku 8.5

	$SMII_0^T$	$SMII_1^T$	$SMII_2^T$	$SMII_3^T$	$SMII_4^T$	$SMII_5^T$	$SMII_6^T$
SMI_0^T	3/5	2/5	5/5	4/5	5/5	2/5	5/5
SMI_1^T	0/3	0/3	0/3	0/3	0/3	0/3	0/3
SMI_2^T	0/3	0/3	0/3	0/3	0/3	0/3	0/3
SMI_3^T	0/3	0/3	0/3	0/3	0/3	0/3	0/3
SMI_4^T	0/3	0/3	0/3	0/3	0/3	0/3	0/3
SMI_5^T	2/2	1/2	0/2	2/2	2/2	2/2	2/2
SMI_6^T	2/2	2/2	0/2	2/2	2/2	2/2	2/2
SMI_7^T	1/6	0/6	6/6	5/6	3/6	0/6	6/6
SMI_8^T	3/6	2/6	4/6	4/6	5/6	2/6	5/6
SMI_9^T	4/8	2/8	4/8	4/8	4/8	2/8	4/8
SMI_{10}^T	3/11	2/11	6/11	5/11	5/11	2/11	8/11
SMI_{11}^T	3/20	2/20	8/20	5/20	5/20	2/20	10/20

Tabela 10.10: Zawieranie się podsieci t-sieci w podsieciach conADT kolejno dla sieci SM I i SM II.

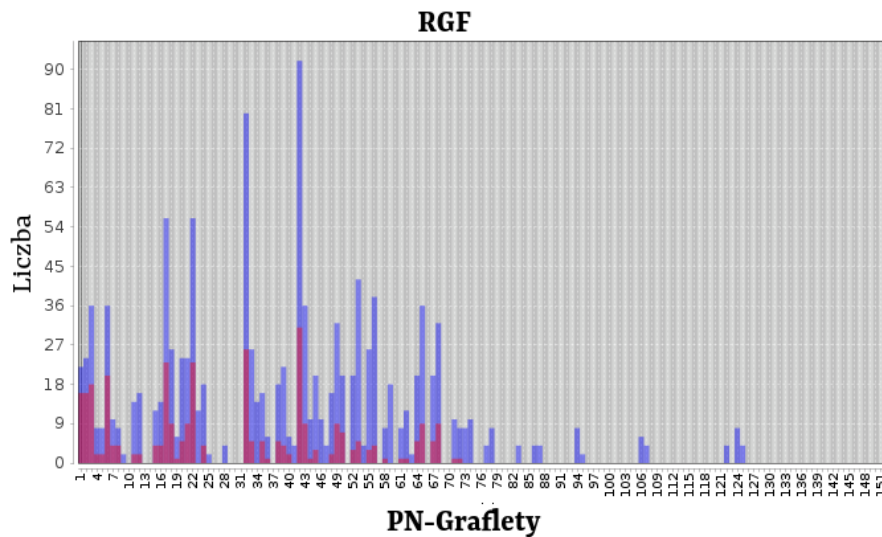
SM I		SM II	
conADT	t-sieci	conADT	t-sieci
0	0,1,7,1	0	0
1	2	1	1
2	3	2	2
3	5,6,8,9,11	3	3
4	4	4	4
		4	5
		4	6

Dokonując podwójnego porównania używając podsieci bazujących na niezmiennikach, ale opisujących podprocesy na różnych poziomach szczegółowości (t-sieci z conADT; t-sieci z MCT; conADT z MCT), możliwe jest sprawdzenie dla części przypadków przypadkowego podobieństwa struktur. W Tabeli 10.10 wskazano, na które z podsieci t-sieci dzielą się podsieci conADT. Dla sieci II wszystkie przypadki conADT są izomorficzne z dokładnie jedną podsiecią t-sieci. Przyglądając się dwóm najlepszym dopasowaniom conADT 0-6 i 3-4 widać, że poziom podobieństwa określony przy conADT po przejściu na mniejsze podsieci t-sieci uległ zmianie. Pojawiły się natomiast częściowe dopasowania mniejszych struktur, które można zaklasyfikować jako przypadkowe podobieństwo.

10.5 Protokoły komunikacji BGP i ECMA

Modele przedstawione w tej sekcji nie reprezentują biologicznych procesów, a przypadki z zastosowań informatycznych i reprezentacji protokołów internetowych. Celem przedstawienia danych modeli w tym rozdziale jest pokazanie możliwości opracowanych metod poza obszarem, dla którego zostały pierwotnie opracowane.

10.5.1 Grafilety



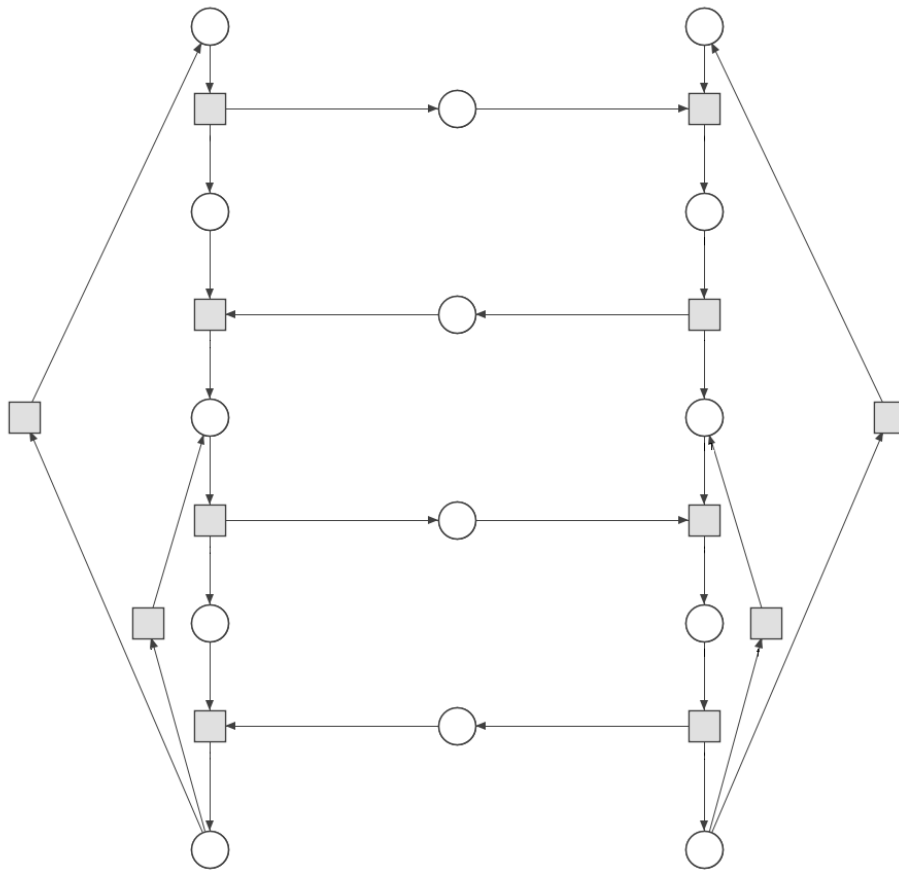
Rysunek 10.5: Rozkład grafiletów w sieciach modelujących protokoły BGP i ECMA.

Wartość RGF równa 158,45 wraz z informacją na temat wielkości porównywanych sieci (kolejno 26 i 30 wierzchołków), pozwala określić, że sieci znacząco różnią się strukturalnie. Poszerzając wiedzę na temat podobieństwa o wyniki cząstkowe z Rysunku 10.5, można stwierdzić, że sieć z Rysunku 10.3 ma prostszą strukturę niż sieć z Rysunku 10.4. Powodem takiego wniosku jest to, że żaden z grafiletów występujących w sieci modelującej protokół BGP nie występuje w niej częściej niż w sieci ECMA (dla większości przypadków mowa o kilkukrotnie mniejszej liczbie wystąpień pn-grafiletów niż w drugim modelu). Dodatkowo w sieci nie występują żadne grafilety z zakresu $G_{75} - G_{84}$ posiadające wierzchołek czwartego stopnia, co oznacza, że model BGP nie posiada żadnego wierzchołka o stopniu 4, w przeciwieństwie do modelu ECMA.

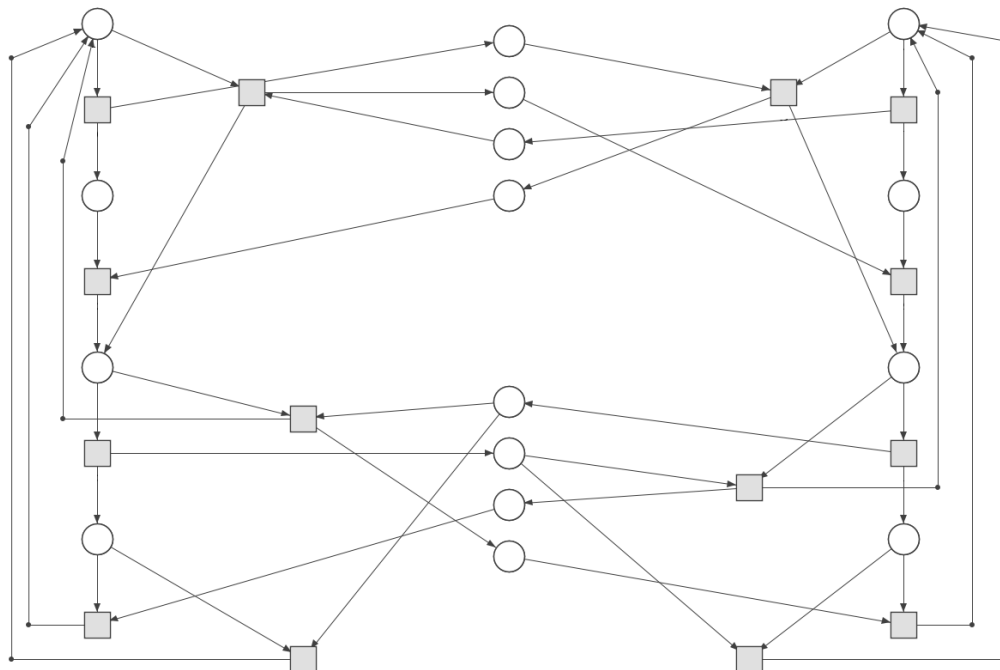
Czy na podstawie powyższych wyników można założyć, że model BGP zawiera się w modelu ECMA? Odpowiedź brzmi nie, natomiast na podstawie grafiletów można stwierdzić, że nie ma strukturalnych dowodów, iż taka sytuacja nie zachodzi.

10.5.2 Dekompozycja do t-sieci i podsieci funkcyjnych

Oba modele dekomponują się do kilku podsieci t-sieci – 4 w przypadku sieci BGP i 8 w przypadku sieci ECMA. Porównanie przeprowadzono dla następujących parametrów: dopasowanie wierzchołków o identycznym stopniu; brak porównań ścieżek t-t z t-p; pętle porównywane tylko z pętlami, spójne podsieci. Wyniki porównań tych podsieci przedstawiono w Tabeli 10.11. Na ich podstawie można wydzielić trzy grupy.



Rysunek 10.3: Model protokołu BGP



Rysunek 10.4: Model protokołu ECMA

Tabela 10.11: Wynik porównań dekomponowanych podsieci t-sieci BGP^T i $ECMA^T$ dla modeli zaprezentowanych na Rysunku 10.3 i 10.4

	$ECMA_0^T$	$ECMA_1^T$	$ECMA_2^T$	$ECMA_3^T$	$ECMA_4^T$	$ECMA_5^T$	$ECMA_6^T$	$ECMA_7^T$
BGP_0^T	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
BGP_1^T	0/3	0/3	0/3	0/3	0/3	0/3	0/3	0/3
BGP_2^T	2/12	2/12	2/12	2/12	8/12	8/12	5/12	5/12
BGP_3^T	2/16	2/16	2/16	2/16	8/16	8/16	5/16	5/16

Pierwsza zawiera podsieci, dla których nie zidentyfikowano żadnego dopasowania na podstawie ich struktury. Są to porównania podsieci BGP_0^T i BGP_1^T z modelu BGP, z podsieciami od $ECMA_0^T$ do $ECMA_7^T$ (odpowiadające im pozycje w tabeli zaznaczono kolorem czerwonym). Wskazane sieci z pierwszego modelu odpowiadają trywialnym sieciom przepływowym, natomiast przypadki drugiego modelu reprezentują proste podsieci zbudowane wokół pojedynczej tranzycji rozgałęziającej. Dla przyjętych parametrów opisane powyżej przypadki nie są uznawane za podobne.

Druga grupa przedstawia częściowe dopasowanie o rozmiarze powyżej 50% (zaznaczone w tabeli kolorem zielonym). Są to porównania podsieci BGP_2^T i BGP_3^T z modelu BGP, z podsieciami od $ECMA_4^T$ do $ECMA_5^T$. Wykazują one podobieństwo oscylujące w przedziale 50 – 60%, co biorąc pod uwagę rozmiar sieci i samych podsieci nadal może stanowić przypadkowe podobieństwo struktur.

Trzecia grupa przedstawia pozostałe wyniki reprezentujące wspólną podsieć o rozmiarze poniżej 50% największej z porównywanych podsieci.

Tabela 10.12: Wynik porównań dekomponowanych podsieci funkcyjnych BGP^F i $ECMA^F$ dla modeli zaprezentowanych na Rysunku 10.3 i 10.4

	$ECMA_0^F$	$ECMA_1^F$	$ECMA_2^F$	$ECMA_3^F$	$ECMA_4^F$	$ECMA_5^F$	$ECMA_6^F$	$ECMA_7^F$
BGP_0^F	3/4	2/4	3/4	2/4	3/4	2/4	3/4	2/4
BGP_1^F	6/8	4/8	5/8	5/8	6/8	4/8	5/8	5/8
BGP_2^F	3/4	2/4	3/4	2/4	3/4	2/4	3/4	2/4
BGP_3^F	3/4	3/4	2/4	2/4	3/4	3/4	2/4	2/4
BGP_4^F	3/4	3/4	2/4	2/4	3/4	3/4	2/4	2/4
BGP_5^F	7/8	3/8	6/8	5/8	7/8	3/8	6/8	5/8
BGP_6^F	3/4	3/4	2/4	2/4	3/4	3/4	2/4	2/4
BGP_7^F	3/4	2/4	3/4	2/4	3/4	2/4	3/4	2/4

Dodatkowo przeprowadzone zostało porównanie z użyciem podsieci funkcyjnych. W wyniku dekompozycji dla obu modeli powstała większa liczba mniejszych podsieci niż w przypadku dekompozycji do t-sieci. W Tabeli 10.12 zielonym kolorem zaznaczono dopasowania uzyskane z użyciem metody węgierskiej zawierające największe sumaryczne podobieństwo. Co należy zaznaczyć, żadne dopasowanie nie jest pełnym. Częściowe dopasowania z wyłączeniem przypadku BGP_0^F i $ECMA_7^F$ wykazują dopasowanie na poziomie powyżej 50%, jednak to, że reprezentują małe choć nietrywialne struktury z dużym prawdopodobieństwem, wskazuje na przypadkowe podobieństwo struktur.

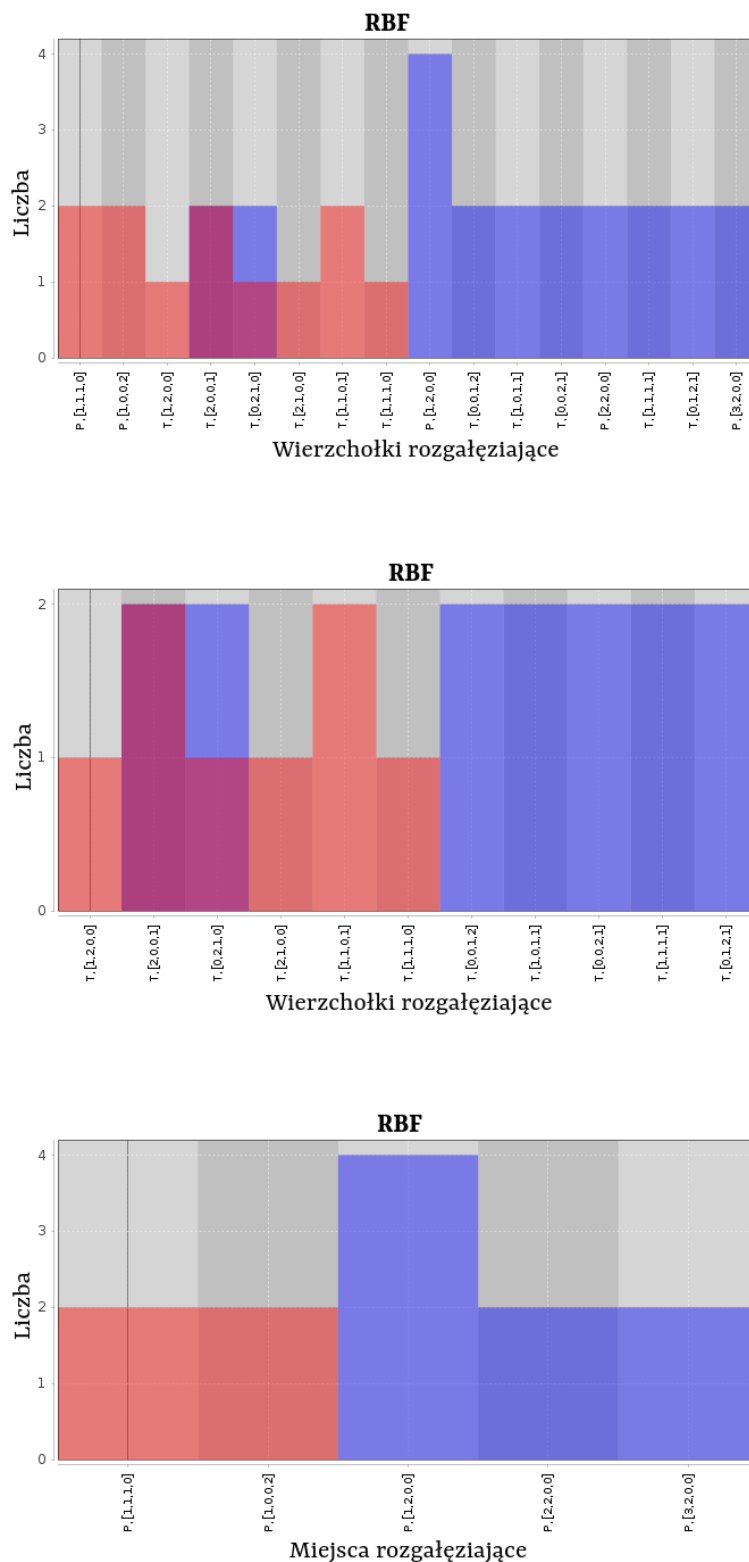
10.5.3 Relative Branching Vertices Frequency distance – RBF

Dystans RBF pomiędzy porównywanymi modelami wynoszący 32,01 wskazuje na znaczące różnice w strukturach (dla porównania przykład z 10.2.2 reprezentował podobny dystans dla większych sieci NER). Różnicą wobec innych przypadków użycia RBF jest dysproporcja pomiędzy wartościami dla tranzycji i miejsc rozgałęziających, wynosząca kolejno 18,047 i 4,852. Ma to swoje źródło w liczbie

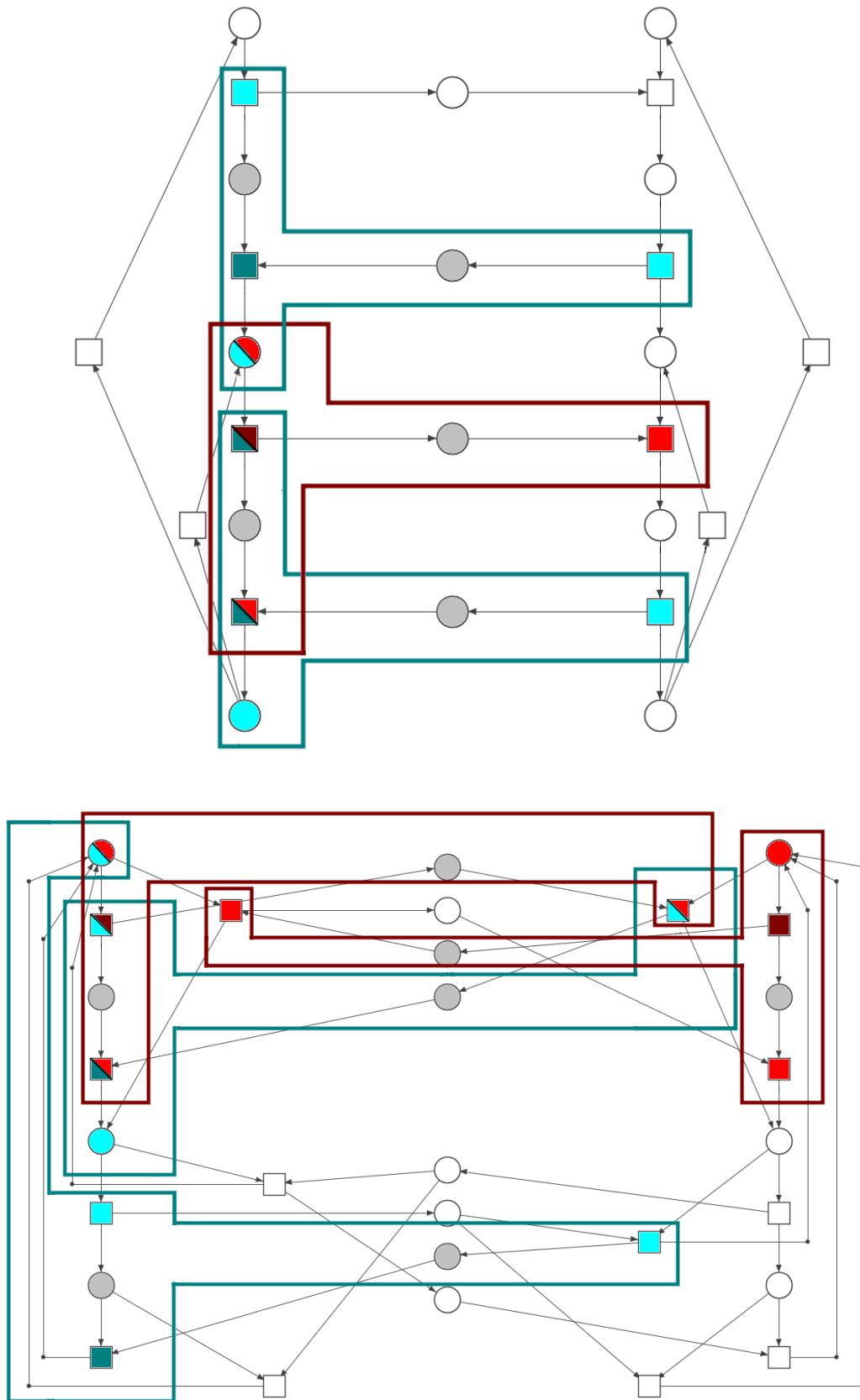
poszczególnych typów wierzchołków – miejsc rozgałęziających w obu sieciach jest prawie 3-krotnie mniej niż tranzycji. Przechodząc do podobieństw, które można ustalić na bazie wyników cząstkowych widocznych na wykresach z Rysunku 10.7, tylko trzy tranzycje rozgałęziające wraz z ich końcówkami mają swoje odpowiedniki w obu sieciach. Przykłady ich lokalizacji w modelach zostały przedstawione na Rysunku 10.6.

Tabela 10.13: Wyniki cząstkowe RBF dla porównania BGP i ECMA

Wierzchołek Rozgałęziający	Cząstkowy RBF
T, [1,2,0,0]	2,485
T, [2,0,0,1]	0,606
T, [0,2,1,0]	0,087
T, [2,1,0,0]	2,485
T, [1,1,0,1]	1,792
T, [1,1,1,0]	2,485
T, [1,1,1,1]	2,398
T, [0,1,2,1]	2,398
T, [0,0,1,2]	2,398
T, [1,0,1,1]	2,398
T, [0,0,2,1]	2,398
P, [1,2,0,0]	1,705
P, [2,2,0,0]	2,398
P, [3,2,0,0]	2,398
P, [1,1,1,0]	1,792
P, [1,0,0,2]	1,792



Rysunek 10.7: Rozkład wierzchołków rozgałęziających w modelach BGP i ECMA



Rysunek 10.6: Lokalizacja tranzycji rozgałęziających występujących w modelach protokołów BGP i ECMA. Niebieski – $T, [2, 0, 0, 1]$, Czerwony – $T, [0, 2, 1, 0]$

10.6 Ewaluacja metod

Metody porównywania sieci Petriego przedstawione w Rozdziałach 6 - 9 przedstawiają znacząco różniące się podejścia do oceny skali podobieństwa. W zależności od rozmiarów porównywanej pary sieci i ich charakterystyk, różne metody będą sugerowane jako efektywne narzędzie do wyznaczenia skali podobieństwa.

10.6.1 Problem różnych źródeł

Porównując modele opisujące zbliżone procesy, ale uzyskane z różnych źródeł (np. pochodzące od różnych autorów lub pobrane z różnych baz danych), można zaobserwować kilka powtarzających się problemów. W zależności od porównywanych sieci możliwe jest równoległe zaistnienie kilku z nich jednocześnie. Są to:

- 1) różny poziom szczegółowości porównywanych procesów;
- 2) styl modelowania reprezentacji autora;
- 3) niejednolity zapis etykiet.

Na bazie trzech powyższych przypadków dokonano poniżej krótkiej charakterystyki każdej z metod porównywania sieci Petriego zaproponowanej w ramach niniejszej rozprawy.

Metoda opisana w Rozdziale 6 – pn-graflety

Graflety reprezentują niewielkie podsieci, a metryki bazujące na nich opierają się w znacznej mierze na ich liczbie. W efekcie każde nadmiarowe (z perspektywy jednej sieci) wystąpienie grafletu (lub jego orbity) jest traktowane jako różnica, bez możliwości określenia, czy powodem jest inny poziom szczegółowości, styl, czy faktyczna różnica biologiczna. Jednocześnie analizując rozkład grafletów możliwe jest wykrycie nadreprezentacji pewnych podsieci, które mogą stanowić przesłankę o różnicach w stylu modelowania autorów. Sytuacja ta zajdzie jednak tylko wtedy, gdy podobieństwo modelowanych procesów zostało potwierdzone z użyciem innej metody (lub źródła).

Metryki grafletowe dokonują pomiaru strukturalnego podobieństwa modeli bez uwzględniania etykiet wierzchołków. Rozszerzenie o ich identyfikację pozbawiłoby te metryki ich głównych zalet, czyli szybkości porównywania.

Metoda opisana w Rozdziale 8 – Wierzchołki rozgałęziające

Podejście bazujące na wierzchołkach rozgałęziających zostało stworzone, aby pomijało różnice w postaci ścieżek przepływowych, które są częstym przypadkiem w sieciach systemów biologicznych. Tworzy to odporność na część różnic strukturalnych mających swoje źródło w różnym poziomie szczegółowości modeli, jak i stylu autora, ale nie na wszystkie występujące przypadki. Wiele przypadków można zidentyfikować i wyeliminować z użyciem grafu zawierania (opisanego pod koniec Rozdziału 8), wspartego wiedzą o redukcjach które można przeprowadzić na podsieciach zawierających wierzchołki rozgałęziające.

Rozróżnianie wierzchołków rozgałęziających może zostać rozszerzone o porównanie etykiet korzenia, jak i jego końcówek. Ze względu na to, że liczba wierzchołków rozgałęziających nie będzie nigdy większa od liczby wierzchołków, nie wpłynie to w istotny sposób na złożoność obliczeniową algorytmu.

Metoda opisana w Rozdziale 7 – Niezmienniki

Metoda porównywania oparta o dopasowywanie niezmienników jest podatna na różnice wynikające z różnych poziomów abstrakcji reprezentowanych modeli, jak i stylu modelowania autorów, które wpływają na liczbę niezmienników jak i na ich postać.

Metoda porównywania oparta o dopasowywanie niezmienników jako jedyna z prezentowanych została stworzona bazując na identyfikacji etykiet wierzchołków. W pierwotnym wariantcie rozróżnienie jest oparte o Komisyjny Numer Enzymu który został przypisany każdemu z wierzchołków, co stanowi efektywne podejście. Jednak w przypadku zastosowania do obszarów spoza sieci metabolicznych pojawia się istotny problem identyfikacji etykiet. Jest on dodatkowo zwielokrotniany przez brak szeroko uznanych standaryzacji nazw.

Metoda opisana w Rozdziale 9 – Dekompozycja

Dokonując porównania zdekomponowanych podsieci nie jest możliwe wykrycie różnic wynikających z różnego poziomu szczegółowości. Zastosowanie innej metody dekompozycji, opisującej bardziej lub mniej szczegółowy podproces, skupi się na różnicach strukturalnych w ramach podsieci. Natomiast zbadanie relacji pomiędzy podsieciami z różnych poziomów szczegółowości i porównanie ich z odpowiednikami innych typów w drugiej sieci może pozwolić określić ich interakcje (np. podział na podsieci t-sieci w sieci reprezentującej proces na wyższym poziomie abstrakcji może odpowiadać podziałowi do conADT bardziej szczegółowo zamodelowanej sieci, przy czym różnice w samych podsieciach pozostaną).

Część różnic wynikających ze stylu modelowania stosowanego przez autora może zostać pominięta przez zmianę parametrów sterujących algorytmem, co przekłada się na postać końcową wyznaczanej wspólnej podsieci.

Algorytm w swojej podstawowej postaci nie dokonuje porównań etykiet pomijając je przy wyznaczaniu podobieństwa strukturalnego. Natomiast możliwe jest przeprowadzenie identyfikacji etykiet w sposób opisany wcześniej dla metody opartej o niezmienniki, przejmując wszystkie wady i zalety tego podejścia.

Wpływ redukcji na uzyskiwane wyniki porównań

Kwestie różnego poziomu szczegółowości porównywanych sieci i stylów modelowania ich autorów mogą zostać rozwiązane w wielu przypadkach z użyciem redukcji sieci. Proces ten, mimo że dostępny i potencjalnie wzmacniający możliwości detekcyjne każdej z opisywanych metod, jest obciążony własnymi ograniczeniami, zwłaszcza umiejętnościami i wiedzą użytkownika na temat modeli, których dotyczy unifikacja i redukcja. Niedokładnie przeprowadzone redukcje mogą prowadzić do fałszywych podobieństw lub różnic.

10.6.2 Problem rozmiaru sieci

- 4) duży rozmiar sieci (>200 wierzchołków);
- 5) mały rozmiar sieci (<50 wierzchołków);
- 6) sieci różnych rozmiarów.

Grafony

Grafony zostały stworzone do operowania na bardzo dużych modelach sieci, o rozmiarach zazwyczaj niespotykanych dla sieci Petriego składających się z dziesiątek tysięcy wierzchołków. Stanowią w tych przypadkach często jedyną alternatywę do wyznaczenia podobieństwa strukturalnego sieci.

W przypadku porównywania małych sieci metryki oparte o pn-grafony charakteryzują się wysoką wrażliwością na bardzo małe różnice. W efekcie uzyskany wynik jest trudny w interpretacji, co w przypadku istniejących alternatyw, które efektywnie operują na małych sieciach, w zasadzie eliminuje grafony z praktycznych zastosowań.

W przypadku porównania sieci o znacząco różnych rozmiarach, metryki oparte o grafony będą nadmiarowe elementy jednej z sieci uznawać za standardowe różnice strukturalne, nie określając czy pomiędzy sieciami nie zachodzi relacja zawierania.

Wierzchołki rozgałęziające

Metoda oparta o wierzchołki rozgałęziające efektywnie radzi sobie przy porównywaniu dużych (powyżej 200 wierzchołków) modeli systemów biologicznych wyrażonych za pomocą sieci Petriego, jak i modeli niewielkich rozmiarów (poniżej 50 wierzchołków). Mimo, że w przeciwieństwie do grafetów porównaniu nie podlegają wszystkie wierzchołki wchodzące w skład sieci, metoda skupia się na kluczowych elementach sieci wyrażonych w postaci wierzchołków rozgałęziających. Jednak dokładność porównania podobnie jak w przypadku porównań z użyciem pn-grafetów jest mniejsza w porównaniu z innymi metodami np. porównania zdedkomponowanych sieci.

Przy porównaniu sieci o znacząco różnych rozmiarach, zachodzi podobna sytuacja jak dla metryk opartych o grafony, z tą różnicą, że dystans jest wyznaczany tylko na bazie wierzchołków rozgałęziających. Dla części przypadków metryka RBF da możliwość wykrycia istniejących podobieństw strukturalnych, ukrytych pośród dużej liczby ścieżek przepływowych.

Niezmienniki

Rozmiar sieci nie jest głównym czynnikiem wpływającym na liczbę istniejących w niej niezmienników. Z tego powodu metoda oparta o niezmienniki może być efektywnie zastosowana dla modeli zbudowanych na kilku tysiącach wierzchołków, a jednocześnie dla sieci o stu wierzchołkach, które zawierają około miliona niezmienników być całkowicie nieefektywnym podejściem do wyznaczania podobieństwa.

Dekompozycja

Efektywność metod opartych o dekompozycje przy porównywaniu dużych modeli nie jest bezpośrednio powiązana z rozmiarem sieci. Elementem kluczowym jest struktura sieci oraz wybrany typ dekompozycji. Przekłada się to na rozmiar i liczbę uzyskanych podsieci oraz często konieczność zachowania

kompromisu pomiędzy nimi. Na przykład operowanie na licznych trywialnych lub bardzo małych podsieciach może dostarczać niewystarczających informacji o podobieństwie. Z drugiej strony, jeśli w wyniku dekompozycji powstaną duże podsieci, to wyznaczenie dla nich wspólnych podsieci może nadal stanowić złożony problem obliczeniowy, którego starano się uniknąć przez użycie dekompozycji.

Porównanie modeli o różnych rozmiarach nie stanowi dodatkowego problemu dla metod opartych o dekompozycje.

10.6.3 Problem wykładniczego wzrostu liczby cząstkowych wektorów

Wspominany na kartach tej rozprawy wykładniczy wzrost liczby cząstkowych wektorów, obserwowany przy wyznaczaniu niezmienników dla niektórych modeli, uniemożliwia wyznaczenie zbioru minimalnych niezmienników a pośrednio dostęp do części metod analitycznych. Mowa tu o dekompozycjach bazujących na niezmiennikach (conADT, conADP, MCT, MCP) oraz o metodach porównujących niezmienniki i t-/s-komponenty. Jednocześnie dla części przypadków, dla których możliwe było wyznaczenie niezmienników, użycie wymienionych metod będzie trudne obliczeniowo, ze względu na dużą liczbę znalezionych niezmienników.

W takich przypadkach możliwe jest skorzystanie z pozostałych metod porównywania. Zwłaszcza możliwość dekompozycji do t-sieci pozwala na uzyskanie wyników zbliżonych do nieosiągalnych conADT czy MCT.

Użycie metod opartych o graflety zwróci informacje na temat strukturalnego podobieństwa, nie uwzględniając przepływów obrazowanych przez niezmienniki. Zastosowanie wierzchołków rozgałęziających dostarczy informacji o kluczowych elementach tworzących niezmienniki i komponenty, jednak bez istotnych informacji na temat skomplikowanych interakcji pomiędzy nimi.

Rozdział 11

Podsumowanie

Problem porównywania grafów i sieci istnieje w przestrzeni naukowej od przynajmniej pół wieku, w wyniku czego powstały całe rodziny metod opisujących na różne sposoby skalę podobieństwa czy różnic występujących pomiędzy parą porównywanych grafów lub sieci. Również w naukach biologicznych prężnie rozwijających się na przestrzeni ostatnich dziesięcioleci, co owocuje licznymi modelami procesów, wzrasta zapotrzebowanie na dedykowane metody porównywania dla sieci Petriego.

Mimo dużego zainteresowania sieciami Petriego w przeciągu ostatnich lat kwestia ich porównywania była bardzo mało eksplorowana. Brak odpowiedniego oprogramowania do analizy sieci Petriego w latach 90-tych, wraz ze stosunkowo niedużą liczbą dostępnych modeli wyrażonych za pomocą sieci Petriego tłumaczy początkowy brak zainteresowania tą kwestią. Jednak w drugiej dekadzie XXI wieku wspomniane ograniczenia przestały stanowić istotny problem, co sprawiło, że brak opracowanych wcześniej dedykowanych metod porównywania pozbawiał użytkowników istotnego narzędzia analitycznego. To właśnie świadomość tej luki i konieczność jej choć częściowego zbadania stanowiły motywację do prac nad tym tematem.

W niniejszej rozprawie doktorskiej badane były różne metody porównywania grafów i sieci, na bazie których w wyniku adaptacji lub inspiracji powstały nowe metody porównywania sieci Petriego. Zmodyfikowano i rozszerzono możliwości techniki porównywania bazującej na niezmiennikach, która do czasu ukazania się artykułów [123, 120, 121] stanowiących podstawę tej rozprawy, była jedyną opisaną metodą dedykowaną do wyznaczania podobieństwa między sieciami Petriego. Stworzono nowe graflety dedykowane do dwudzielnej, skierowanej, wielołukowej struktury odpowiadającej sieciom Petriego. Jednocześnie przetestowano istniejące metryki grafletowe, oceniając ich możliwości i preferowane sytuacje zastosowań. Opracowano nowe metody porównywania, od początku tworzone z uwzględnieniem charakterystyki sieci Petriego. Jednym z rezultatów jest rodzina metod opartych na wyznaczaniu wspólnych podsieci w zdekomponowanych modelach procesów biologicznych. Poświęcono tutaj szczególną uwagę możliwości uzyskania struktur o znaczeniu biologicznym i ich modyfikacji w zależności od kontekstu biologicznego, w którym operuje metoda. Została ona dostosowana do działania na różnych typach zdekomponowanych podsieci, skupiając się na grupie powiązanej z niezmiennikami, będącymi głównym narzędziem analizy biologicznej przeprowadzanej za pomocą sieci Petriego. W rezultacie analiz struktur i charakterystyk sieci Petriego opracowana została definicja wierzchołków rozgałęziających, stanowiąca podstawę dla algorytmu porównywania przez dekompozycję, która jednocześnie wyewoluowała do postaci samodzielnej metody. Wszystkie z opisywanych metod były rozważane w kontekście systemów biologicznych i szeroko pojętej biologii. Ten obszar nauki nie jest

wolny od problemów modelowania procesów znanych z innych obszarów nauki, czy przemysłu. Kwestia interpretacji różnic, pomijania nieistotnych przypadków, czy też unifikacji modeli w celu uzyskania rzeczywistego pomiaru podobieństwa przewija się przez całą pracę, odnosząc się do różnych perspektyw prezentowanych przez kolejne metody porównywania.

Niniejsza praca powinna być traktowana jako pierwszy duży krok w procesie analizy struktur sieci Petriego przez ich wzajemne porównywanie, jak i kolejny mały krok z perspektywy teorii sieci Petriego czy teorii grafów. Istnieje wciąż wiele metod porównywania wspomnianych w trzecim rozdziale, które mają potencjał do adaptacji i efektywnego działania dla modeli wyrażonych za pomocą sieci Petriego. Algorytmy prezentowane w tej rozprawie, w większości przypadków zostały opracowane dla średnich rozmiarów modeli wyrażonych za pomocą sieci Petriego, gdzie głównym celem przyświecającym pracy było sprawdzenie jakości zwracanych wyników. Autor rozprawy jest świadomy, że niektóre z przedstawionych algorytmów posiadają obszary podatne na optymalizację w celu redukcji ogólnej złożoności obliczeniowej. Przykładem może być kwestia znalezienia wszystkich pn-grafletów w sieci, która w opisanej postaci opiera się na podejściu stosowanym dla grafów nieskierowanych i nie wykorzystuje potencjału dwudzielności sieci Petriego do optymalizacji.

W ramach niniejszej pracy zostały osiągnięte następujące cele:

- zebrano i przeanalizowano różne metody porównywania grafów pod względem ich adaptacji do porównania sieci biologicznych przedstawionych za pomocą sieci Petriego;
- do operowania na sieciach Petriego dostosowano metody grafletowe wraz z opracowaniem nowych struktur (pn-grafletów) oraz gruntownymi testami metryk RGF i GDDA w celu określenia ich możliwości w nowym środowisku;
- opracowano nowe metody porównywania sieci Petriego. Pierwsza bazuje na dekompozycji do podsieci, w których znajdowana jest wspólna podsieć o znaczeniu biologicznym. Druga jest metodą bazująca na znajdowaniu rozkładu wierzchołków rozgałęziających;
- przeprowadzono testy mające określić zalety i ograniczenia prezentowanych metod;
- opracowano wytyczne do zastosowania poszczególnych metod porównywania i interpretacji ich wyników.

W porównywaniu struktur dwóch modeli biologicznych wyrażonych za pomocą sieci Petriego kluczowe jest zrozumienie, że interpretacja wyników może znacznie się różnić w zależności od kontekstu modelowania i podejścia badacza do klasyfikacji elementów struktury jako wspólnych lub różnych. W rezultacie koniecznym jest, aby dostępna była paleta metod porównywania reprezentujących różne podejścia do oceny stopnia podobieństwa jak i dedykowanych do sieci Petriego różnych rozmiarów, oraz spełniających różne charakterystyki.

Rezultaty uzyskane za pomocą metod prezentowanych w ramach niniejszej rozprawy dowodzą ich użyteczności przy porównywaniu modeli systemów biologicznych. Jednocześnie, mimo że od początku były tworzone one dla bardzo konkretnego działu nauki, mogą być łatwo zaadaptowane do efektywnego porównywania modeli z innych dziedzin, gdzie sieci Petriego są stosowane.

Bibliografia

- [1] ADEREM, A. Systems biology: its practice and challenges. *Cell* 121, 4 (2005), 511–513.
- [2] AHMED, N. K., NEVILLE, J., ROSSI, R. A., AND DUFFIELD, N. Efficient graphlet counting for large networks. *2015 IEEE international conference on data mining* (2015), 1–10.
- [3] ALI, W., RITO, T., REINERT, G., SUN, F., AND DEANE, C. M. Alignment-free protein interaction network comparison. *Bioinformatics* 30, 17 (2014), i430–i437.
- [4] AMSTEIN, L. K., ACKERMANN, J., HANNIG, J., ĐIKIĆ, I., FULDA, S., AND KOCH, I. Mathematical modeling of the molecular switch of tnfr1-mediated signaling pathways using Petri nets. *bioRxiv* (2021).
- [5] APARICIO D, RIBEIRO P, S. F. Extending the Applicability of Graphlets to Directed Networks. *IEEE/ACM Trans Comput Biol Bioinform* 14 (11-12 2017), 1302–1315.
- [6] ARAÚJO, M., AND ROQUE, L. Modeling games with Petri nets. *Digra conference* (2009).
- [7] BABAI, L. Graph isomorphism in quasipolynomial time. *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing* (2016), 684–697.
- [8] BALAZ, V., KOCA, J., KVASNICKA, V., AND SEKANINA, M. A metric for graphs. *Casopis pro pestovani matematiky* 111, 4 (1986), 431–433.
- [9] BALDAN, P., COCCO, N., DE NES, F., SEGURA, M. L., AND SIMEONI, M. Mpath2pn-translating metabolic pathways into Petri nets. *BioPPN2011 Int. Workshop on Biological Processes and Petri Nets, CEUR Workshop Proceedings 724* (2011), 102–116.
- [10] BALDAN, P., COCCO, N., AND SIMEONI, M. Comparison of metabolic pathways by considering potential fluxes. *Biological Processes & Petri Nets* (2012).
- [11] BALDAN, P., COCCO, N., AND SIMEONI, M. Representing and comparing metabolic pathways as Petri nets with mpath2pn and cometa. *Electronic Notes in Theoretical Computer Science* 299 (2013), 5–13.
- [12] BARATÈ, A., HAUS, G., AND LUDOVICO, L. A. Music analysis and modeling through Petri nets. *Computer Music Modeling and Retrieval: Third International Symposium, CMMR 2005, Pisa, Italy, September 26-28, 2005. Revised Papers 3* (2006), 201–218.
- [13] BARNETT, J. H. Early writings on graph theory: Euler circuits and the königsberg bridge problem. *Part II: Historical Projects in Discrete Mathematics and Computer Science of Resources*

- for Teaching Discrete Mathematics: Classroom Project, History Modules, and Articles*, MAA Notes 74 (2005), 197–208.
- [14] BAUSE, F., AND KRITZINGER, P. S. *Stochastic Petri nets*, vol. 1. Vieweg Wiesbaden, 2002.
 - [15] BERTHELOT, G. Transformations and decompositions of nets. *Advanced Course on Petri Nets* (1986), 359–376.
 - [16] BERTHOMIEU, B., AND VERNADAT, F. Time Petri nets analysis with tina. *QEST 6* (2006), 123–124.
 - [17] BEST, E., CHERKASOVA, L., AND DESEL, J. Compositional generation of home states in free choice systems. *Annual Symposium on Theoretical Aspects of Computer Science* (1991), 398–409.
 - [18] BHUIYAN, M. A., RAHMAN, M., RAHMAN, M., AND AL HASAN, M. Guise: Uniform sampling of graphlets for large graph analysis. *2012 IEEE 12th International Conference on Data Mining* (2012), 91–100.
 - [19] BONNICI, V., GIUGNO, R., PULVIRENTI, A., SHASHA, D., AND FERRO, A. A subgraph isomorphism algorithm and its application to biochemical data. *BMC bioinformatics 14*, 7 (2013), 1–13.
 - [20] BONNICI, V., GIUGNO, R., PULVIRENTI, A., SHASHA, D., AND FERRO, A. A subgraph isomorphism algorithm and its application to biochemical data. *BMC bioinformatics 14*, 7 (2013), 1–13.
 - [21] BRAUER, W., AND REISIG, W. Carl adam Petri and "Petri nets". *Informatik-Spektrum 29*, 5 (2006), 369–374.
 - [22] BUNKE, H. On a relation between graph edit distance and maximum common subgraph. *Pattern recognition letters 18*, 8 (1997), 689–694.
 - [23] CAMPBELL, D. M., AND RADFORD, D. Tree isomorphism algorithms: Speed vs. clarity. *Mathematics Magazine 64*, 4 (1991), 252–261.
 - [24] CHAOUIYA, C. Petri net modelling of biological networks. *Briefings in bioinformatics 8*, 4 (2007), 210–219.
 - [25] CHAOUIYA, C., REMY, E., AND THIEFFRY, D. Petri net modelling of biological regulatory networks. *Journal of Discrete Algorithms 6*, 2 (2008), 165–177.
 - [26] CORNEIL, D. G., AND GOTLIEB, C. C. An efficient algorithm for graph isomorphism. *Journal of the ACM (JACM) 17*, 1 (1970), 51–64.
 - [27] CORNEIL, D. G., AND KIRKPATRICK, D. G. A theoretical analysis of various heuristics for the graph isomorphism problem. *SIAM Journal on Computing 9*, 2 (1980), 281–297.
 - [28] DALE, M. R. *Graph Structure and System Function: Graphlet Methods*. Cambridge University Press, 2017, p. 252–270.

- [29] DAVID, R., AND ALLA, H. *Discrete, continuous, and hybrid Petri nets*, vol. 1. Springer, 2005.
- [30] DESEL, J., AND REISIG, W. The concepts of Petri nets. *Software & Systems Modeling* 14 (2015), 669–683.
- [31] EINLOFT, J., ACKERMANN, J., NÖTHEN, J., AND KOCH, I. Monalisa—visualization and analysis of functional modules in biochemical networks. *Bioinformatics* 29, 11 (2013), 1469–1470.
- [32] EMMERT-STREIB, F., DEHMER, M., AND SHI, Y. Fifty years of graph matching, network alignment and network comparison. *Information sciences* 346 (2016), 180–197.
- [33] ESPARZA, J., AND SILVA, M. Compositional synthesis of live and bounded free choice Petri nets. *International Conference on Concurrency Theory* (1991), 172–187.
- [34] FLORIN, G., AND NATKIN, S. Les reseaux de Petri stochastiques; theorie, techniques de calcul, applications. *Thèses de doctorat d'état, Univ. Paris VI* (1985).
- [35] FORMANOWICZ, D., GUTOWSKA, K., SZAWULAK, B., AND FORMANOWICZ, P. The crosstalk between sars-cov-2 infection and the raa system in essential hypertension—analyses using systems approach. *International journal of molecular sciences* 22, 19 (2021), 10518.
- [36] FORMANOWICZ, D., RADOM, M., AND FORMANOWICZ, P. Modelowanie udziału żelaza w powstawaniu miażdżycy-podejście systemowe. *Kosmos* 63, 3 (2014), 331–344.
- [37] FUJITA, S., MATSUNO, H., NAGASAKI, M., MIYANO, S., ET AL. Constructing biological pathway models with hybrid functional Petri nets. *In silico biology* 4, 3 (2004), 271–291.
- [38] GAREY, M. R., AND JOHNSON, D. S. *Computers and intractability*, vol. 174. freeman San Francisco, 1979.
- [39] GIUA, A., AND SILVA, M. Petri nets and automatic control: A historical perspective. *Annual Reviews in Control* 45 (06 2018).
- [40] GOMES, L. *Recent Advances in Petri Nets Modeling*. Multidisciplinary Digital Publishing Institute, 2022.
- [41] GONEN, M., AND SHAVITT, Y. Approximating the number of network motifs. *Internet Mathematics* 6, 3 (2009), 349–372.
- [42] GRIMALDI, R. P. *Discrete and Combinatorial Mathematics, 5-th edition*. Pearson Education India, 2006.
- [43] GROHE, M., AND NEUEN, D. Recent advances on the graph isomorphism problem. *arXiv preprint arXiv:2011.01366* (2020).
- [44] GUTOWSKI, Ł., GUTOWSKA, K., PIORUŃSKA-STOLZMANN, M., FORMANOWICZ, P., AND FORMANOWICZ, D. Systems approach to study associations between oxldl and abdominal aortic aneurysms. *International Journal of Molecular Sciences* 20, 16 (2019), 3909.
- [45] HACK, M. H. T. *Petri net language*. Technical Report, 1976.

- [46] HARARY, F. *Graph theory addison-wesley reading ma usa*. Wiley New York, 1969.
- [47] HARARY, F. *Graph Theory*. Taylor & Francis In, 1994.
- [48] HARTMANIS, J., AND BERMAN, L. On isomorphisms and density of np and other complete sets. *Proceedings of the eighth annual ACM symposium on Theory of computing* (1976), 30–40.
- [49] HASAN, A., CHUNG, P.-C., AND HAYES, W. Graphettes: Constant-time determination of graphlet and orbit identity including (possibly disconnected) graphlets up to size 8. *PloS one* 12, 8 (2017), e0181570.
- [50] HAYES, W., SUN, K., AND PRŽULJ, N. Graphlet-based measures are suitable for biological network comparison. *Bioinformatics* 29, 4 (2013), 483–491.
- [51] HEINER, M. Understanding network behavior by structured representations of transition invariants. *Algorithmic Bioprocesses* (2009), 367–389.
- [52] HEINER, M., GILBERT, D., AND DONALDSON, R. Petri nets for systems and synthetic biology. *Formal Methods for Computational Systems Biology: 8th International School on Formal Methods for the Design of Computer, Communication, and Software Systems, SFM 2008 Bertinoro, Italy, June 2-7, 2008 Advanced Lectures 8* (2008), 215–264.
- [53] HEINER, M., RICHTER, R., SCHWARICK, M., AND ROHR, C. Snoopy-a tool to design and execute graph-based formalisms. *Petri Net Newsletter* 74 (2008), 8–22.
- [54] HELFGOTT, H. A., BAJPAI, J., AND DONA, D. Graph isomorphisms in quasi-polynomial time. *arXiv preprint arXiv:1710.04574* (2017).
- [55] HOČEVAR, T., AND DEMŠAR, J. A combinatorial approach to graphlet counting. *Bioinformatics* 30, 4 (2014), 559–565.
- [56] HOHEISEL, A., AND ALT, M. *Petri nets*. Springer, 2007.
- [57] HOPCROFT, J. E., AND TARJAN, R. E. Av log v algorithm for isomorphism of triconnected planar graphs. *Journal of Computer and System Sciences* 7, 3 (1973), 323–331.
- [58] HOU, C. L., MA, Z., AND QI, X. Z. Research on completeness in decomposition and composition of Petri nets. *Applied Mechanics and Materials* 190 (2012), 297–303.
- [59] HOČEVAR, T., AND DEMŠAR, J. Combinatorial algorithm for counting small induced graphs and orbits. *PLOS ONE* 12, 2 (02 2017), 1–17.
- [60] JENSEN, K. Coloured Petri nets and the invariant-method. *Theoretical computer science* 14, 3 (1981), 317–336.
- [61] JENSEN, K. High-level Petri nets. *Applications and Theory of Petri Nets: Selected Papers from the 3rd European Workshop on Applications and Theory of Petri Nets Varenna, Italy, September 27–30, 1982 (under auspices of AFCET, AICA, GI, and EATCS)* (1983), 166–180.
- [62] KADEN, F. Graphmetriken und distanzgraphen. *ZKI-Informationen, Akad. Wiss. DDR* 2, 82 (1982), 1–63.

- [63] KADEN, F. Graph similarity and distances. *Topics in Combinatorics and Graph Theory*, Physica-Verlag (1990), 397–404.
- [64] KANEHISA, M. The kegg database. *‘In Silico’Simulation of Biological Processes: Novartis Foundation Symposium 247 247* (2002), 91–103.
- [65] KANEHISA, M., ARAKI, M., GOTO, S., HATTORI, M., HIRAKAWA, M., ITOH, M., KATAYAMA, T., KAWASHIMA, S., OKUDA, S., TOKIMATSU, T., ET AL. Kegg for linking genomes to life and the environment. *Nucleic acids research* 36, suppl_1 (2007), D480–D484.
- [66] KELMANS, A. K. Comparison of graphs by their number of spanning trees. *Discrete Mathematics* 16, 3 (1976), 241–261.
- [67] KOCH, I., AND ACKERMANN, J. On functional module detection in metabolic networks. *Metabolites* 3, 3 (2013), 673–700.
- [68] KOCH, I., NÖTHEN, J., AND SCHLEIFF, E. Modeling the metabolism of arabidopsis thaliana: application of network decomposition and network reduction in the context of Petri nets. *Frontiers in genetics* 8 (2017), 85.
- [69] KOGUT, D., GUTOWSKA, K., POTERALA-HEJMO, A., SMIEJA, J., FORMANOWICZ, D., AND FORMANOWICZ24, P. Petri nets and ode as complementary tools in analysis of signaling pathways. *Proceedings of 11th International Conference on Bioinformatics and Computational Biology* 60 (2019), 150–160.
- [70] KONIEWSKI, R. Rozproszona symulacja stochastycznych sieci petriego. *XV Krajowa Konferencja Automatyki* 29, 5 (2006), 369–374.
- [71] KURATOWSKI, K. Sur le problème des courbes gauches en topologie. *Fund. Math.* 15 (1930), 271–283.
- [72] KWONG, Y. On reduction of asynchronous systems. *Theoretical Computer Science* 5, 1 (1977), 25–50.
- [73] LEE, K.-H., AND FAVREL, J. Hierarchical reduction method for analysis and decomposition of Petri nets. *IEEE transactions on Systems, Man, and Cybernetics*, 2 (1985), 272–280.
- [74] LEE-KWANG, H., FAVREL, J., AND BAPTISTE, P. Generalized Petri net reduction method. *IEEE transactions on systems, man, and cybernetics* 17, 2 (1987), 297–303.
- [75] LIPETS, V., VANETIK, N., AND GUDES, E. Subsea: an efficient heuristic algorithm for subgraph isomorphism. *Data Mining and Knowledge Discovery* 19 (2009), 320–350.
- [76] LIPTON, R. J. Reduction: a new method of proving properties of systems of processes. *Proceedings of the 2nd ACM SIGACT-SIGPLAN symposium on Principles of programming languages* (1975), 78–86.
- [77] LIU, F., AND HEINER, M. Colored Petri nets to model and simulate biological systems. *Recent advances in Petri Nets and concurrency* 827 (2012), 71–85.

- [78] LUKS, E. M. Isomorphism of graphs of bounded valence can be tested in polynomial time. *Journal of computer and system sciences* 25, 1 (1982), 42–65.
- [79] MATERI, W., AND WISHART, D. S. Computational systems biology in drug discovery and development: methods and applications. *Drug discovery today* 12, 7-8 (2007), 295–303.
- [80] MCGREGOR, J. J. Backtrack search algorithms and the maximal common subgraph problem. *Software: Practice and Experience* 12, 1 (1982), 23–34.
- [81] MERLIN, P. M. *A Study of the Recoverability of Computing Systems*. University of California, Irvine, 1974.
- [82] MILO, R., ITZKOVITZ, S., KASHTAN, N., LEVITT, R., SHEN-ORR, S., AYZENSHTAT, I., SHEFFER, M., AND ALON, U. Superfamilies of evolved and designed networks. *Science* 303, 5663 (2004), 1538–1542.
- [83] MURATA, T. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE* 77, 4 (1989), 541–580.
- [84] NG, K. M., REAZ, M. B. I., AND ALI, M. A. M. A review on the applications of Petri nets in modeling, analysis, and control of urban traffic. *IEEE Transactions on Intelligent Transportation Systems* 14, 2 (2013), 858–870.
- [85] NOULEHO ILEMO, S., BARTH, D., DAVID, O., QUESSETTE, F., WEISSER, M.-A., AND WATEL, D. Improving graphs of cycles approach to structural similarity of molecules. *PloS one* 14, 12 (2019), e0226680.
- [86] OOTSUKI, J. T., FUJII, Y., MIZUTANI, H., AND SEKIGUCHI, T. Immune system derived approach for finding firing sequences of a sub-class of Petri nets. *IEEJ Transactions on Electronics, Information and Systems* 118, 3 (1998), 419–427.
- [87] OOTSUKI, J. T., KATO, T., AND SEKIGUCHI, T. An immune system response network approach for finding firing sequences of a sub-class of Petri nets. *SMC'98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No. 98CH36218)* 1 (1998), 148–153.
- [88] OOTSUKI, J. T., AND SEKIGUCHI, T. Application of the immune system reaction mechanism concept to sequential control. *1999 7th IEEE International Conference on Emerging Technologies and Factory Automation. Proceedings ETFA'99 (Cat. No. 99TH8467)* 2 (1999), 1391–1396.
- [89] OYELADE, J., ISEWON, I., OGUNBONA, O., AROMOLARAN, O., AND SOYEMI, J. Modeling of metabolic pathways using Petri net. *2017 International Conference on Computational Science and Computational Intelligence (CSCI)* (2017), 1276–1283.
- [90] PAPADIMITRIOU, C. H. *Computational complexity*. Addison-Wesley, 1994.
- [91] PAVLENKO, O., SHRAMENKO, N., AND MUZYLYOV, D. Logistics optimization of agricultural products supply to the european union based on modeling by Petri nets. *New Technologies, Development and Application III* 6 (2020), 596–604.

- [92] PELEG, M., RUBIN, D., AND ALTMAN, R. B. Using Petri net tools to study properties and dynamics of biological systems. *Journal of the American Medical Informatics Association* 12, 2 (2005), 181–199.
- [93] PETERSON, J. L. Petri nets. *ACM Computing Surveys (CSUR)* 9, 3 (1977), 223–252.
- [94] PETRI, C. A. Communication with automata. Supplement I RADC-TR-65-377, Rome Air Development Center, Research and Technology Division, Air Force Systems Command, Griffiss Air Force Base, New York, January 1966. Volume I Final Report, Distribution of this document is unlimited.
- [95] PROCK, J. A new technique for fault detection using Petri nets. *Automatica* 27, 2 (1991), 239–245.
- [96] PRŽULJ, N. Biological network comparison using graphlet degree distribution - errata. *Bioinformatics* 23, 2 (2007), e177–e183.
- [97] PRŽULJ, N., CORNEIL, D. G., AND JURISICA, I. Efficient estimation of graphlet frequency distributions in protein–protein interaction networks. *Bioinformatics* 22, 8 (2006), 974–980.
- [98] PRŽULJ, N. Biological network comparison using graphlet degree distribution. *Bioinformatics* 23, 2 (01 2007), e177–e183.
- [99] PRŽULJ, N., CORNEIL, D. G., AND JURISICA, I. Modeling interactome: scale-free or geometric? *Bioinformatics* 20, 18 (07 2004), 3508–3515.
- [100] RADOM, M. (prywatna korespondencja). *Instytut Informatyki, Politechnika Poznańska*.
- [101] RADOM, M., AND FORMANOWICZ, P. Extended time Petri nets. *arXiv preprint arXiv:2405.09208* (2024).
- [102] RADOM, M., MACHNICKA, M. A., KRZAWICZ, J., BUJNICKI, J. M., AND FORMANOWICZ, P. Petri net–based model of the human dna base excision repair pathway. *PloS one* 14, 9 (2019), e0217913.
- [103] RADOM, M., RYBARCZYK, A., SZAWULAK, B., ANDRZEJEWSKI, H., CHABELSKI, P., KOZAK, A., AND FORMANOWICZ, P. Holmes: a graphical tool for development, simulation and analysis of Petri net based models of complex biological systems. *Bioinformatics* 33, 23 (2017), 3822–3823.
- [104] RAMCHANDANI, C. *Analysis of asynchronous concurrent systems by timed Petri nets*. 1974.
- [105] RAY, L.C., K. R. Finding chemical records by digital computers. *Science* 126 (1957).
- [106] REDDY, V. N., MAVROVOUNIOTIS, M. L., LIEBMAN, M. N., ET AL. Petri net representations in metabolic pathways. *ISMB* 93 (1993), 328–336.
- [107] RITO, T., WANG, Z., DEANE, C. M., AND REINERT, G. How threshold behaviour affects the use of subgraphs for network comparison. *Bioinformatics* 26, 18 (2010), i611–i617.
- [108] SACKMANN, A., HEINER, M., AND KOCH, I. Application of petri net based analysis techniques to signal transduction pathways. *BMC bioinformatics* 7, 1 (2006), 1–17.

- [109] SANFELIU, A., AND FU, K.-S. A distance measure between attributed relational graphs for pattern recognition. *IEEE transactions on systems, man, and cybernetics*, 3 (1983), 353–362.
- [110] SARAJLIĆ ANIDA, MALOD-DOGNIN NOEL, Y. O. N. P. N. Graphlet-based characterization of directed networks. *Scientific Reports* 6, 35098 (2016), 1–14.
- [111] SCHÄRER, O. D. Nucleotide excision repair in eukaryotes. *Cold Spring Harbor perspectives in biology* 5, 10 (2013), a012609.
- [112] SCHEIDEL, J., LINDAUER, K., ACKERMANN, J., AND KOCH, I. Quasi-steady-state analysis based on structural modules and timed Petri net predict system’s dynamics: The life cycle of the insulin receptor. *Metabolites* 5, 4 (2015), 766–793.
- [113] SCHWARICK, M. A software to to analyse Petri net models. *Diploma thesis* (2009).
- [114] SERRATOSA, F. Redefining the graph edit distance. *SN Computer Science* 2, 6 (2021), 438.
- [115] SILVA, M. 50 years after the phd thesis of carl adam Petri: A perspective. *IFAC Proceedings Volumes* 45, 29 (2012), 13 – 20. 11th IFAC Workshop on Discrete Event Systems.
- [116] SKORUPSKI, J. Sieci Petriego jako narz dzie do modelowania procesów ruchowych w transporcie. *Prace Naukowe Politechniki Warszawskiej, Transport (ISSN 1230-9265)*, 78 (2011), 69–77.
- [117] SOBIK, F. Graphmetriken und klassifikation strukturierter objekte. *ZKI-Informationen, Akad. Wiss. DDR* 2, 82 (1982), 63–122.
- [118] SORENSON, T. A method of establishing groups of equal amplitude in plant sociology based on similarity of species content, and its application to analysis of vegetation on danish commons. *Kong Dan Vidensk Selsk Biol Skr* 5 (1948), 1–5.
- [119] SUSSENGUTH, E. H. A graph-theoretic algorithm for matching chemical structures. *Journal of Chemical Documentation* 5, 1 (1965), 36–43.
- [120] SZAWULAK, B., AND FORMANOWICZ, P. Graphlets in comparison of Petri net-based models of biological systems. *Scientific Reports* 12, 1 (2022), 20942.
- [121] SZAWULAK, B., AND FORMANOWICZ, P. Comparison by partition - finding Petri nets similarity on the basis of subnets. *W recenzji* (2024).
- [122] SZAWULAK, B., AND FORMANOWICZ, P. Graphlet degree distribution sensibility in Petri net with low density. *W recenzji* (2024).
- [123] SZAWULAK BARTŁOMIEJ, F. P. Dekompozycja sieci jako podstawa do porównywania sieci Petriego. *Automatyzacja Procesów Dyskretnych* (2018).
- [124] TORÁN, J., AND WAGNER, F. The complexity of planar graph isomorphism. *Bull. EATCS* 97 (2009), 60–82.
- [125] TRARES, K., ACKERMANN, J., AND KOCH, I. The canonical and non-canonical nf- κ b pathways and their crosstalk: A comparative study based on Petri nets. *Biosystems* 211 (2022), 104564.

- [126] ULLMANN, J. R. An algorithm for subgraph isomorphism. *Journal of the ACM (JACM)* 23, 1 (1976), 31–42.
- [127] VIZING, V. G. Some unsolved problems in graph theory. *Russian Mathematical Surveys* 23, 6 (1968), 125.
- [128] WEBB, EDWIN C, N.-I. *Enzyme nomenclature 1992. Recommendations of the Nomenclature Committee of the International Union of Biochemistry and Molecular Biology on the Nomenclature and Classification of Enzymes*. No. Ed. 6. Academic Press, 1992.
- [129] WEGRZYN, M. Petri net decomposition approach for partial reconfiguration of logic controllers. *IFAC Proceedings Volumes* 39, 17 (2006), 323–328.
- [130] WIEBKING, D. Graph isomorphism in quasipolynomial time parameterized by treewidth. *arXiv preprint arXiv:1911.11257* (2019).
- [131] WIŚNIEWSKI, R., STEFANOWICZ, Ł., BUKOWIEC, A., AND LIPÍŃSKI, J. Theoretical aspects of Petri nets decomposition based on invariants and hypergraphs. *Multimedia and ubiquitous engineering* (2014), 371–376.
- [132] XUE, R. Testing isomorphism of graphs in polynomial time. *arXiv preprint arXiv:2305.12688* (2023).
- [133] YANG, C.-C., AND MAY, C. P. Algorithms for finding directed graph isomorphisms by finite automata. *International Journal of Computer & Information Sciences* 9 (1980), 117–140.
- [134] YAO, F. F. *Graph 2-isomorphism is NP-complete*. STANFORD UNIV., COMPUTER SCIENCE Department, SCHOOL OF HUMANITIES AND SCIENCES, 1979.
- [135] YAVEROĞLU, Ö. N., MALOD-DOGNIN, N., DAVIS, D., LEVNAJIC, Z., JANJIC, V., KARPANDZA, R., STOJMIROVIC, A., AND PRŽULJ, N. Revealing the hidden language of complex networks. *Scientific reports* 4, 1 (2014), 4547.
- [136] YAVEROĞLU, Ö. N., MILENKOVIĆ, T., AND PRŽULJ, N. Proper evaluation of alignment-free network comparison methods. *Bioinformatics* 31, 16 (2015), 2697–2704.
- [137] ZAITSEV, D. Decomposition of Petri nets. *Cybernetics and Systems Analysis* 40, 5 (2004), 739–746.
- [138] ZAITSEV, D. *Clans of Petri Nets: Verification of protocols and performance evaluation of networks*. 06 2013.
- [139] ZAITSEV, D., AND SLEPTSOV, A. I. State equations and equivalent transformations for timed Petri nets. *Cybernetics and Systems Analysis* 33, 5 (1997), 659–672.
- [140] ZAITSEV, D. A. Decomposition-based calculation of Petri net invariants. *Proceedings of Workshop on Token based computing of the 25-th International conference on application and theory of Petri nets, Bologna, Italy* (2004), 79–83.

- [141] ZELINKA, B. On a certain distance between isomorphism classes of graphs. *Casopis pro pestovani matematiky* 100, 4 (1975), 371–373.
- [142] ZELINKA, B. Distance between graphs. *Fourth Czechoslovakian Symposium on Combinatorics, Graphs and Complexity* (1992).
- [143] ZENG, Q. Two symmetrical decomposition methods for structure-complex Petri nets and their applications. *Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD 2007) 3* (2007), 1101–1106.
- [144] ZENG, Z., TUNG, A. K., WANG, J., FENG, J., AND ZHOU, L. Comparing stars: On approximating graph edit distance. *Proceedings of the VLDB Endowment* 2, 1 (2009), 25–36.