

ZACHODNIOPOMORSKI UNIWERSYTET TECHNOLOGICZNY
W SZCZECINIE
WYDZIAŁ INFORMATYKI

mgr inż. Łukasz MAZUROWSKI
STRESZCZENIE ROZPRAWY DOKTORSKIEJ

Model systemu algorytmicznego komponowania muzyki na podstawie wzorców

Promotor rozprawy:
prof. dr hab. inż. Antoni WILIŃSKI

Recenzenci rozprawy:
prof. dr hab. inż. Dariusz KANIA
Wydział Automatyki, Elektroniki i Informatyki
Politechnika Śląska

dr hab. inż. Przemysław KLĘSK
Wydział Informatyki
Zachodniopomorski Uniwersytet Technologiczny w Szczecinie

Szczecin, 2017

Spis treści

1. Aktualność problemu	2
2. Przedmiot badań	3
3. Metody naukowe stosowane podczas wykonywania badań	4
4. Główny cel rozprawy	5
5. Zadania do wykonania	5
6. Wartość teoretyczna	6
7. Wartość praktyczna	6
8. Akceptacja wyników przez społeczność	6
9. Główne osiągnięcia pracy	7
10.Struktura i układ pracy	7
11.Koncepcje algorytmicznego komponowania muzyki	9
11.1. Komponowanie bez definiowania formy generowanego utworu	11
11.2. Komponowanie uwzględniające opis formy generowanego utworu	17
12.Spis publikacji własnych	26
13.Podsumowanie	26
Literatura	29

1. Aktualność problemu

Systemy algorytmicznego komponowania muzyki można pogrupować, uwzględniając rodzaj reguł komponowania, wykorzystanych w procesie generowania muzyki. Wg [47] systemy te można podzielić na dwie grupy. Pierwszą stanowią rozwiązania z regułami jawnymi, w których kompozytor definiuje z góry zasady komponowania utworu. Drugą grupę stanowią systemy z regułami niejawnymi, w których reguły komponowania pozyskiwane są z przykładów (fragmentów bądź całych utworów muzycznych) za pomocą wybranych metod przetwarzania i pozyskiwania informacji. W tab. 1 przedstawiono klasyfikację systemów algorytmicznego komponowania muzyki analizowanych przez autora pod względem jawności i niejawności reguł komponowania zastosowanych w algorytmie generującym muzykę.

W nawiązaniu do przedstawienia różnych podejść z dziedziny algorytmicznego komponowania muzyki zasadne wydaje się wskazanie pewnej luki w prezentowanych rozwiązaniach jaką jest tzw. muzyka tła (*background music*), będąca jednym z najbardziej interesujących obszarów stosowania muzyki użytkowej. Prezentowane w autorskiej dysertacji koncepcje z założenia funkcjonują jako narzędzia generujące muzykę tła, która odnosi się do *“muzyki jakiegokolwiek rodzaju, która jest odtwarzana podczas danej czynności, w taki sposób, aby człowiek aktywnie nie angażował się w słuchanie tej muzyki”* (na podstawie [24]). Ten typ muzyki odgrywa drugorzędną rolę w życiu człowieka, jednak według różnych badań ma wpływ na wiele aspektów naszego zachowania.

Badania prowadzone przez Kämpfe i innych w [36] skupione były na analizie wpływu muzyki na różne zadania wykonywane przez człowieka. Autorzy zasugerowali, że chociaż globalne wyniki analizy nie wykazują znaczącego wpływu na efektywność zadań, to szczegółowe badania pozwalają jednak zidentyfikować zarówno pozytywne jak i negatywne oddziaływanie na analizowane czynności. Cockerton i inni w [15] śledzili pozytywny i negatywny wpływ słuchania muzyki podczas badania funkcji poznawczych (rozwiązywania testu kognitywnego). Także Chaudhury i inni w [12] zaprezentowali rolę muzyki w procesie przeprogramowania aktywności mózgu. Autorzy wywnioskowali, że muzyka może wywoływać różne mechanizmy funkcjonowania mózgu m.in. proces nauczania oraz pamiętania.

Przedstawione analizy są jedynie częścią badań prowadzonych na świecie, mających znaczenie nie tylko poznawcze ale również komercyjne. Autor przypuszcza, że system generujący muzykę tła adekwatną do zadanego rodzaju towarzyszącej jej czynności, byłby narzędziem użytecznym rynkowo, pozwalającym na generowanie odpowiedniego gatunku muzyki w zależności od aktualnych potrzeb klienta-słuchacza. Na bazie analizy istniejących rozwiązań opisanych w literaturze oraz niewypełnionej luki badawczej związanej z systemami algorytmicznego komponowania muzyki tła, powstały koncepcje systemów algorytmicznego komponowania muzyki na podstawie wzorców.

Autor podejmując opisywaną w tej pracy problematykę kierował się, oprócz własnymi fascynacjami, także a może i przede wszystkim aktualnością prowadzonych badań. Ośrodki takie jak *MIT*, *Columbia University* [51], *University College London* [37] czy *Griffith University* [52], oferując w ramach prowadzonych kursów tematykę związaną z muzyką generatywną, kształtują kierunek rozwoju tej gałęzi nauki. W odczuciu autora dorobek badawczy przedstawiony w tej pracy pozostaje w zgodzie z

tym narastającym trendem, niemniej jednak pozostawiając w dalszym ciągu szerokie pole do doskonalenia, rozszerzania i poprawiania uzyskanych autorskich metod tworzenia muzyki generatywnej.

Rozwiązanie opisane w	Bazuje na	Wykorzystanie jawnych regułach	Wykorzystanie niejawnych regułach
[38]	automaty komórkowe	X	-
[7]	gramatyki	X	X
[3], [4], [5]	algorytm genetyczny	X	-
[43]	algorytm genetyczny	X	-
[45]	algorytm genetyczny	X	-
[32]	algorytm genetyczny	X	-
[10]	algorytm roju cząstek (PSO)	X	-
[33]	algorytm mrówkowy	X	-
[41], [50]	wirtualny model systemu chemicznego	X	-
[55], [54]	sztuczne sieci neuronowe	X	-
[42]	drzewa decyzyjne	-	X
[28], [29]	parsowanie przyrostowe i drzewa PST	-	X
[17]	modele statystyczne	-	X
[25]	ukryte modele Markowa (HMM)	-	X
[30]	łańcuchy Markowa	-	X
[53]	łańcuchy Markowa	-	X
[13], [27]	sztuczne sieci neuronowe	-	X
[19], [18], [20], [21], [22], [16]	sieci ATN i MATN	X	X

Tab. 1: Podział systemów algorytmicznego komponowania muzyki ze względu na jawność i niejawnosć wykorzystanych reguł komponowania. Opracowanie własne.

2. Przedmiot badań

Przedmiotem badań jest opracowanie autorskich koncepcji algorytmicznego komponowania muzyki (akceptowalnej przez człowieka i być może użytecznej rynkowo), w których opracowano autorskie metody przetwarzania treści muzycznych, bazujące na metodach sztucznej inteligencji. W pracy wykorzystane zostały metody probabilistyczne (np. generowanie progresji akordów) jak i eksploracji danych (estymacja tonacji wzorca, rozkład klas długości dźwięków wzorca, okienkowanie wzorca w celu generowania melodii i linii basowej).

3. Metody naukowe stosowane podczas wykonywania badań

Podczas opracowania metod algorytmicznego komponowania muzyki na podstawie wzorców wykorzystano metody naukowe opisane w tab. 2.

Metoda	Wykorzystana w celu
metoda analizy literatury	wskazania kierunku dyskusji naukowej i wykazania luki badawczej
metoda estymacji tonacji Krumhansl-Kessler'a [14]	oszacowania tonacji wzorca
model Parncutt'a czasowych akcentów dźwięku [46]	obliczenia histogramu klas wysokości dźwięków wzorca
metoda statystyczna (rozkład zmiennej czasu trwania dźwięków wzorca)	obliczenia histogramu klas długości dźwięków wzorca
obserwacja zmiany wartości wysokości dźwięku	analizy wpływu zmian wartości parametru na postać wynikowej kompozycji muzycznej
obserwacja zmiany wartości długości wzorca	analizy wpływu zmian wartości parametru na postać wynikowej kompozycji muzycznej
obserwacja zmiany parametrów zbiorów wygenerowanych i wykorzystanych wzorców	analizy wpływu zmian wartości parametru na postać wynikowej kompozycji muzycznej
obserwacja zmiany struktury tablicy możliwych powtórzeń wzorca w kompozycji	analizy wpływu zmian wartości parametru na postać wynikowej kompozycji muzycznej
obserwacja zmiany struktury opisującej formę kompozycji	analizy wpływu zmian wartości parametru na postać wynikowej kompozycji muzycznej
obserwacja zmiany struktury opisującej skalę muzyczną	analizy wpływu zmian wartości parametru na postać wynikowej kompozycji muzycznej
obserwacja zmiany struktury opisującej lokalizację dźwięków w obrębie wzorca	analizy wpływu zmian wartości parametru na postać wynikowej kompozycji muzycznej
obserwacja zmiany struktury opisującej przesunięcia (offsety) dźwięków w obrębie wzorca	analizy wpływu zmian wartości parametru na postać wynikowej kompozycji muzycznej
obserwacja zmiany liczby i czasu trwania akordów w obrębie wzorca	analizy wpływu zmian wartości parametrów na postać wynikowej kompozycji muzycznej

Tab. 2: Metody naukowe stosowane podczas wykonywania badań. Opracowanie własne.

4. Główny cel rozprawy

Celem pracy jest opracowanie modelu komputerowego systemu komponowania muzyki użytkowej (posiadającej wartość rynkową), w którym do generowania oryginalnych utworów muzycznych wykorzystuje się metody autorskie oparte na metodach sztucznej inteligencji.

Po analizie dotychczasowych rozwiązań w zakresie algorytmicznego komponowania utworów oraz konstruowania maszyn do generowania muzyki sztucznej a także na podstawie badań autorskich algorytmów stawia się hipotezę, że, bazując na metodach sztucznej inteligencji, możliwe jest opracowanie takich algorytmów komponowania utworów muzycznych, których efektem będą utwory atrakcyjne w odbiorze i posiadające strukturę poprawną w aspekcie teorii muzyki. W powyższej hipotezie badawczej zakłada się, że system komponowania bazował będzie na wzorcowych utworach stanowiących muzyczną bazę wiedzy oraz na metodach wyszukiwania i modyfikacji sygnatur muzycznych (komponentów utworu), wykorzystywanych w kompozycji wynikowej. Przez model w ujęciu metodologicznym autor rozumie narzędzie kognitywne umożliwiające badanie i opisywanie rozmaitych aspektów struktury i funkcjonalności systemu komponowania muzyki. Model taki będzie krokiem pośrednim do syntezy metody lub metod generowania muzyki sztucznej prowadzącej w przyszłości do budowania złożonego systemu informatycznego. Przez system ten autor rozumie z kolei rozbudowaną instalację sprzętowo-programową, której budowa nie jest celem niniejszej pracy. Praca ta może być inspiracją, pierwszym krokiem do budowy i doskonalenia takiego systemu.

W przypuszczeniu, że utwory będą atrakcyjne w odbiorze kryje się wiele subiektywizmu i ta cecha utworów będzie wymagała specyficznego potwierdzenia. Generalnie trudno zdefiniować bezwzględną atrakcyjność utworu muzycznego, z pewnością jest to niemożliwe, można jednak posłużyć się miarami pośrednimi – statystycznie mierzonym zainteresowaniem lub rynkowymi np. wartością sprzedaży. Drugi wymieniony istotny postulat – zgodność z przyjętymi kanonami teorii muzyki, jest znacznie prostszy do dowiedzenia i zapewnienia.

5. Zadania do wykonania

W ramach opracowania metod algorytmicznego komponowania muzyki na podstawie wzorców muzycznych wykonano następujące zadania:

1. przeprowadzono analizę istniejących rozwiązań z dziedziny algorytmicznego komponowania muzyki pod kątem wyszukania luki badawczej,
2. zdefiniowano komponenty procesu generowania muzyki tła w ujęciu systemowym,
3. opracowano metodę generowania kompozycji muzycznych na podstawie wzorców bazującą na mini-generatorach,
4. zaimplementowano metodę z pkt. 3 w środowisku Matlab w celu przeprowadzenia eksperymentów,
5. przeprowadzono szereg eksperymentów, których wyniki były analizowane przez autora i upubliczniane w ramach działań popularyzatorskich,
6. na podstawie metody z pkt. 3 oraz w odpowiedzi na uwagi środowiska odbiorców kompozycji zdefiniowano metodę generowania kompozycji na podstawie wzorców, uwzględniającą opis struktury kompozycji wynikowej,

7. zaimplementowano metodę z pkt. 6 w środowisku Matlab w celu przeprowadzenia eksperymentów,
8. empirycznie zweryfikowano hipotezę postawioną w pracy.

6. Wartość teoretyczna

Wartość teoretyczna rozprawy jest wynikiem opracowania dwóch metod algorytmicznego komponowania muzyki na podstawie wzorców, w ramach których dokonano sformalizowania aspektów zadań procesu komponowania muzyki (tab. 3 na str. 10) związanych z:

- generowaniem melodii,
- rozszerzaniem idei melodycznych,
- harmonizacją,
- hybrydowym podejściu do budowy struktury utworu.

Teoretyczną wartością dodaną w tym względzie jest zdefiniowanie formalizmów wzorca muzycznego, mini-generatora oraz kompozycji wyjściowej a także algorytmów:

- generowania melodii na podstawie sekwencji akordów,
- generowania linii basowej na podstawie melodii,
- generowania wzorca muzycznego za pomocą mini-generatora.

7. Wartość praktyczna

Wartość praktyczną dysertacji stanowią liczne przykłady publikacji wygenerowanych kompozycji w ramach różnych wydarzeń, w tym z udziałem mediów. Należy w tym miejscu wspomnieć o prezentowaniu rzeczywistych kompozycji i własnych koncepcji autora na antenie Polskiego Radia Szczecin (audycja „Trącić Myszką” [1], [2]) oraz podczas różnych lokalnych festiwali i wydarzeń z branży IT („rabbIT” oraz „Noc Naukowców”). Podczas wymienionych wydarzeń autor z wykorzystaniem sprzętu multimedialnego oraz prototypu własnego systemu algorytmicznego komponowania muzyki oddziaływał na uczestników pokazów, a także dawał możliwość kreowania własnych koncepcji muzycznych opisując i udostępniając autorskie metody generowania kompozycji. Akcje promowania wyników badań były prowadzone również wśród najmłodszych słuchaczy czego przykładem może być Festiwal Nauki i Sztuki w Stargardzie, podczas którego autor prezentował możliwość algorytmizacji procesu tworzenia muzyki za pomocą komputera [35]. Popularyzowanie autorskiej koncepcji odbywało się również przy okazji wydarzeń kulturalnych takich jak wernisaże fotografii, w których tło muzyczne stanowiły kompozycje utworzone za pomocą autorskich metod generowania muzyki.

8. Akceptacja wyników przez społeczność

Czasopisma i materiały konferencyjne, w których recenzenci oceniali wyniki badań cząstkowych:

- *Metody Informatyki Stosowanej*, Polska Akademia Nauk (2008),
- *Modele inżynierii teleinformatyki*. Wybrane zastosowania 4, Wydawnictwo Uczelniane Politechniki Koszalińskiej (2009),

- *Modele inżynierii teleinformatyki*. Wybrane zastosowania 5, Wydawnictwo Uczel-
niane Politechniki Koszalińskiej (2010),
- *Przegląd Elektrotechniczny*, Wydawnictwo Czasopism i Książek Technicznych
SIGMA-NOT Spółka z o.o (2012),
- *IEEE Xplore Digital Library*, IEEE (2012),
- *Journal of Applied Computer Science*, Politechnika Łódzka (2012),
- *Przegląd Elektrotechniczny*, Wydawnictwo Czasopism i Książek Technicznych
SIGMA-NOT Spółka z o.o (2015),
- *AIP Conference Proceedings* 1648, AIP Publishing (2015).

Konferencje na których zaprezentowano oraz przedyskutowano wyniki badań
częstkowych:

- *Krajowa Konferencja Studentów i Młodych Pracowników Nauki*, Darłowo (2009),
- *Krajowa Konferencja Studentów i Młodych Pracowników Nauki*, Darłowo (2010),
- *International Conference System Modelling and Control*, Łódź (2011),
- *Advanced Computer Systems*, Międzyzdroje (2012),
- *International Symposium on Multimedia Applications and Processing* (w ramach
konferencji *FedCIS*), Wrocław (2012),
- *International Conference on Numerical Analysis and Applied Mathematics*, Ro-
dos (2014).

9. Główne osiągnięcia pracy

Do głównych autorskich osiągnięć pracy należy zaliczyć:

- opracowanie metod algorytmicznego komponowania muzyki na podstawie wzor-
ców w aspekcie muzyki tła, wariancie zakładającym tworzenie kompozycji o
strukturze ciągłej oraz strukturze zbudowanej na podstawie zdefiniowanej formie
opisu jej składowych,
- opracowanie algorytmu generowania wzorców muzycznych na podstawie zdefi-
niowanych parametrów wejściowych dla tego procesu,
- opracowanie algorytmu generowania linii melodycznej na podstawie wejściowej
sekwencji akordów,
- opracowanie algorytmu generowania linii basowej na podstawie danej sekwencji
melodycznej.

10. Struktura i układ pracy

Rozprawa doktorska liczy 195 stron oraz składa się z pięciu rozdziałów a także
glosariusza, spisu rysunków, spisu tablic, załącznika w postaci listingu kodów źró-
dłowych oraz załącznika opisującego zawartość dołączonej do pracy płyty CD. W
rozprawie ulokowano glosariusz zawierający 47 pojęć. Ponadto praca zawiera 58
rysunki, 34 tabele oraz 24 listingi kodów. Bibliografia liczy 108 pozycji.

W rozdziale 1 dokonano teoretycznego wprowadzenia do zagadnienia algo-
rytmicznego komponowania muzyki oraz przeanalizowano historycznie istotne
rozwiązania z tej dziedziny, ujęte w literaturze światowej. Dla potrzeby prawidło-
wego zrozumienia przez Czytelnika problematyki zawartej w pracy, autor dokonał
przeglądu głównych zagadnień związanych z teorią muzyki, w wymiarze wystar-
czającym do przyswojenia treści zawartych w pracy. Podsumowaniem dokonanego

przeglądu rozwiązań i odniesienia do problemu badań jest sformułowanie celu pracy doktorskiej oraz hipotezy badawczej.

W rozdziale 2 przedstawiono koncepcję algorytmicznego komponowania muzyki bez definiowania formy utworu (muzyka tła o strukturze ciągłej) oraz koncepcję uwzględniającą formalny opis struktury kompozycji. Ponadto przedstawiono główne komponenty procesu komponowania muzyki, zaproponowane przez Bilesa w [40] (rozdział 2.1).

Rozdział 2.1 opisuje pierwszy wariant komponowania, który zakłada generowanie kompozycji muzycznej, bazujące na generowaniu wzorców muzycznych i łączeniu ich w jedną kompozycję. Wzorce muzyczne są generowane za pomocą parametrów wejściowych, takich jak długość wzorca, wektor opisujący sekwencję wysokości dźwięków motywu muzycznego, wektor opisujący możliwe rozmieszczenie dźwięków w ramach wzorca, itp. Kompozycja muzyczna tworzona jest natomiast z wygenerowanych wcześniej wzorców, których możliwość połączeń definiuje macierz, utworzona na podstawie analizy właściwości poszczególnych wzorców (ich podobieństwa).

Rozdział 2.2 opisuje drugi wariant komponowania, który bazuje na opisie formy utworu końcowego (liczbie i kolejności łączenia głównych fragmentów utworu) oraz procesie generowania kompozycji, wykorzystującym wygenerowane wcześniej wzorce muzyczne, których łączenie uwzględnia zdefiniowaną wcześniej formę kompozycji. Proces generowania wzorców ma charakter odmienny, ponieważ w odróżnieniu od pierwszego wariantu, głównym elementem nie jest motyw muzyczny, a wygenerowana (na podstawie reguł) progresja akordów, dla której tworzona jest melodia i dalsze partie instrumentalne.

Rozdział 2.3 porusza aspekt doboru instrumentów muzycznych oraz syntezy kompozycji wynikowej za pomocą syntezy programowego.

W rozdziale 3 autor skupił się na prezentacji przykładowej implementacji prototypów koncepcji przedstawionych w rozdziałach 2.2 i 2.3. Rozdział 3.1 stanowi opis kolejnych elementów implementacji koncepcji generowania muzyki o strukturze ciągłej wraz ze szczegółowym opisem kodów źródłowych zawartych w dodatku do pracy. Analogicznie rozdział 3.2 zawiera opis implementacji koncepcji generowania muzyki, uwzględniającej opis jej struktury. W rozdziale 3.3 przedstawiono etap syntezy kompozycji z wykorzystaniem oprogramowania SynthFont.

Eksperymenty i wyniki badań zostały szczegółowo opisane w rozdziale 4. W rozdziale 4.1 autor przedstawił eksperymenty dotyczące generowania muzyki tła bez definiowania formy kompozycji. W ramach badań skupiono się na obserwacji następujących parametrów:

- modyfikacji długości dźwięków muzycznych,
- modyfikacji długości wzorca,
- modyfikacji liczności zbiorów wygenerowanych oraz wykorzystanych w kompozycji wzorców muzycznych,
- modyfikacji struktury zawierającej liczby możliwych powtórzeń wzorca w kompozycji.

Rozdział 4.2 zawiera opis obserwacji następujących parametrów:

- modyfikacji struktury opisującej formę kompozycji,
- modyfikacji struktury opisującej skalę muzyczną wykorzystaną w procesie komponowania,
- modyfikacji struktury opisującej możliwe przesunięcia czasowe dźwięków wchodzących w skład wzorców muzycznych,

- modyfikacji struktury opisującej możliwe długości dźwięków wykorzystane w budowie wzorca,
- modyfikacji liczby i czasu trwania akordów.

W rozdziale 5 zaprezentowano konkluzje oraz perspektywy rozwoju i wdrożeń koncepcji omawianych w dysertacji. Ujęto w nim proces publikowania otrzymanych wyników w aspekcie naukowego rozwoju autora. W obrębie dalszych perspektyw opisano aplikację “Wirtualne Laboratorium Muzykologa” opracowaną przez autora w ramach projektu TEWI [49], która może stanowić wstęp do utworzenia złożonego systemu algorytmicznego komponowania muzyki generatywnej na podstawie wzorców muzycznych pozyskanych z wejściowego zbioru gotowych utworów. We wnioskach końcowych autor dokonał podsumowania obu koncepcji komponowania oraz wykazał aktualność problemu naukowego.

11. Koncepcje algorytmicznego komponowania muzyki

Przed przystąpieniem do opisywania autorskich koncepcji systemów algorytmicznego komponowania muzyki, należy w sposób formalny przedstawić szereg zadań składających się na proces komponowania muzyki, ujęty przez Bileśa w [40] na stronach 34-36 i stanowiący przedmiot badań niniejszej dysertacji. Zadania te zostały podzielone na grupy, których struktura została przedstawiona w tab. 3. Porządek przedstawionych zadań procesu komponowania został przyjęty arbitralnie przez Bileśa i nie odzwierciedla struktury tworzenia muzyki według modelu kaskadowego.

Ważnym aspektem procesu komponowania w systemach algorytmicznego komponowania muzyki jest generowanie motywu melodii (tab. 3). Częstym podejściem jest wykorzystanie generatora melodii, który, bazując na zdefiniowanych bądź wyuczonych regułach generowania, wytwarza krótkie (najczęściej kilku- bądź kilkunastonutowe) fragmenty monofoniczne. Dodatkowo generowanie melodii może zostać rozbite na podzadania dotyczące tworzenia (w sposób niezależny) sekwencji wartości rytmicznych oraz wysokości dźwięków. Niektóre systemy oferują możliwość jednoczesnego generowania sekwencji rytmiczno-częstotliwościowych (np. [3]).

Część systemów algorytmicznego komponowania muzyki generuje jedynie sekwencje rytmiczne, najczęściej o długości jednego taktu, który jest traktowany jako wzorec, a następnie umieszczany i powielany w różnych częściach kompozycji. Systemy takie swoją funkcjonalność opierają na idei tzw. maszyny perkusyjnej (z ang. *drum machine*) - urządzeniu, które przechowuje w pamięci (lub syntezyje w czasie rzeczywistym) próbki dźwięków instrumentów perkusyjnych, wykorzystywanych następnie do tworzenia jedno- bądź wielotaktowych wzorców zapętlnych w celu uzyskania podkładu perkusyjnego. W takich systemach wysokości dźwięku nie są interpretowane jako częstotliwości lecz jako określone identyfikatory instrumentów wykorzystywanych podczas komponowania taktu.[40]

W procesie komponowania kolejnym ważnym zagadnieniem jest rozszerzanie idei melodycznych (tab. 3), które ogólnie można zdefiniować jako serie transformacji motywu początkowego, wykorzystane do generowania motywu docelowego. Zdefiniowane serie transformacji mogą generować wariacje motywu (np. [8], [9], [23]), wzorce rytmiczne ([11]) lub łączyć krótkie motywy w dłuższe linie melodyczne ([9], [8]). Punktem wejściowym w takich systemach jest motyw składający się z kilku lub kilkunastu nut, który jest następnie wykorzystany do tworzenia dłuższej struktury.

Generowanie motywów melodii	Sekwencje wysokości dźwięków bez rytmu; Sekwencje rytmicznych bez określenia wysokości dźwięków; Sekwencje dźwięków z informacją o wysokości i rytmie.
Rozszerzenie idei melodycznych	Generowanie wariacji i melodii kontrapunktowych; Łączenie fragmentów melodii w dłuższe elementy.
Harmonizacja	Generowanie partii harmonii; Generowanie zmian akordów.
Aranżacja	Definiowanie indywidualnych partii dla danego muzyka; Definiowanie instrumentów przypadających na indywidualne partie.
Struktura	Wykorzystywanie wysokiego poziomu opisu (top-down); Budowanie lub ewaluacja struktury od podstaw (bottom-up); Podejście hybrydowe w definiowaniu struktury (top-down i bottom-up).

Tab. 3: Podstawowe zadania procesu komponowania (na podstawie [40]). Opracowanie własne.

Harmonizacja (tab. 3) stanowi ważny element procesu komponowania muzyki i w ramach różnych systemów algorytmicznego komponowania polega najczęściej na tworzeniu głosu sopranowego, altowego, tenorowego oraz basowego dla danej melodii. Trudność polega na tym, że wszystkie wymienione głosy muszą być traktowane jako indywidualne melodie, z jednoczesnym zachowaniem harmonii wertykalnej (harmonii pomiędzy melodią oraz pozostałymi głosami). Można więc potraktować zagadnienie harmonizacji melodii jako proces planowania (problem optymalizacyjny) bazujący na zdefiniowanych wcześniej regułach następstwa dźwięków w czasie oraz ich rozmieszczenia w pionie (wybrzmiewania w tej samej chwili, z zachowaniem zasad harmonii). W przypadku gdy wejściem do systemu jest zarówno melodia (do której ma zostać dobrana harmonizacja) oraz progresja akordów, problem jest relatywnie prosty ([34]). Sytuacja komplikuje się, gdy wejściem systemu komponowania jest jedynie melodia. W takim przypadku stosuje się zazwyczaj metody bazujące na systemie reguł lub sieciach neuronowych, jak np. [44].

Aranżacja (tab. 3) stanowi problem doboru fragmentów utworu do określonych partii instrumentów muzycznych. Częściowo problem ten jest rozwiązywany w systemach rozbudowujących (rozszerzających) melodie oraz harmonizujących wejściowy materiał muzyczny. Istnieją jednak systemy ([9]), w których wyodrębniony został osobny moduł funkcjonalny odpowiedzialny za aranżację (tzw. aranżer).

Ostatnim i prawdopodobnie najtrudniejszym zagadnieniem procesu komponowania muzyki jest tworzenie logicznej struktury utworu (tab. 3). Niektóre systemy algorytmicznego komponowania muzyki poruszają się w płaszczyźnie niskiego poziomu opisu kompozycji (na ogół na poziomie opisu pojedynczej frazy utworu). Istnieją również systemy, które są w stanie generować w pełni rozwinięte kompozycje, poruszając się po różnych płaszczyznach opisu struktury utworu (np. od frazy

po formę) [8]. Systemy takie mają najczęściej charakter interaktywny i pozwalają (lub często wymagają) zdefiniowania przez użytkownika struktury na najwyższym poziomie opisu kompozycji w sposób manualny ([48]) bądź częściowo zautomatyzowany ([8], [9]). Kształtowanie struktury w systemach komponowania jest realizowane również z wykorzystaniem różnych technik i metod obliczeń ewolucyjnych takich jak agenty, automaty komórkowe lub PSO (optymalizacja rojem cząsteczek) ([6], [31]).

Przedstawiona analiza procesu komponowania ujęta przez Bileśa w [40] pozwala autorowi dokonać próby ulokowania swojej koncepcji systemu algorytmicznego komponowania muzyki na tle innych rozwiązań jako systemu, w którym:

- sekwencja melodii generowana jest z uwzględnieniem informacji na temat wysokości i długości dźwięków ją tworzących;
- rozszerzenie melodii polega na łączeniu jej fragmentów w dłuższe elementy, z zastosowaniem zdefiniowanych reguł;
- generowanie harmonii bazuje na regułach zmian akordów związanych, na bazie których tworzona jest melodia wiodąca;
- etap aranżacji ujęty jako zdefiniowanie instrumentów do określonych partii utworu pozostaje pod kontrolą użytkownika;
- struktura utworu definiowana jest z poziomu wysokiej abstrakcji opisu formy kompozycji (top-down), a możliwe połączenia fragmentów utworu w jedną kompozycję wynikają z analizy właściwości tych fragmentów (bottom-up).

11.1. Komponowanie bez definiowania formy generowanego utworu

Generowanie muzyki nieposiadającej skryształizowanej formy (tzn. początku, rozwinięcia i zakończenia) jest zagadnieniem dość typowym dla muzyki tła, której cechą charakterystyczną są powtarzające się fragmenty muzyczne i ich wariacje. Taki sposób generowania muzyki może być porównany do generowania dużej tekstury obrazu z relatywnie małych fragmentów. [39]

W przyjętej koncepcji systemu wyjściowa kompozycja muzyczna tworzona jest z mniejszych fragmentów muzycznych, zwanych wzorcami muzycznymi. Wzorzec jest w tym podejściu zdefiniowany jako macierz nut zaproponowana w [26], której przykład zaprezentowano w tab. 4, przy czym dodatkowo macierz ta posiada stałą i jednakową dla wszystkich wierszy wartość długości dur_W nuty (w jednostkach bpm, i sekundach). Uproszczenie to jest podyktowane sposobem generowania wzorca.

Onset (bpm)	Dur. (bpm)	MIDI ch.	Pitch	Vol.	Onset (sec.)	Dur. (sec.)
0	0.4167	1.00	60.00	100.00	0	0.25
0.4167	0.4167	1.00	62.00	100.00	0.25	0.25
0.8333	0.4167	1.00	64.00	100.00	0.50	0.25
1.2500	0.4167	1.00	65.00	100.00	0.75	0.25
1.6667	0.4167	1.00	67.00	100.00	1.00	0.25
2.0833	0.4167	1.00	69.00	100.00	1.25	0.25
2.5000	0.4167	1.00	71.00	100.00	1.50	0.25
2.9167	0.4167	1.00	72.00	100.00	1.75	0.25

Tab. 4: Przykładowa macierz nut w środowisku Matlab. Opracowanie własne.

Przykładową macierz o rozmiarze $[8 \times 7]$, opisującą oktawę od dźwięku $C4$ (60) do $C5$ (72), przedstawiono w tab. 4. Każdy wiersz macierzy reprezentuje sparаметryzowany dźwięk muzyczny. Kolejne kolumny zawierają wartości parametrów danego dźwięku. W pierwszej kolumnie znajdują się wartości rozpoczęcia dźwięku w jednostkach bpm, związanych z tempem odgrywania utworu. Druga kolumna zawiera wartości długości dźwięków w jednostkach bpm. W kolumnie trzeciej występują numery wykorzystanego przez dany dźwięk kanału *MIDI* (standard *MIDI* zakłada jednocześnie wykorzystanie 16 kanałów, przy czym w przytoczonym przykładzie z tab. 4 wykorzystywany jest jedynie kanał nr 1). Kolumna czwarta zawiera wartości wysokości dźwięków zapisane w postaci liczb *MIDI*, które odpowiadają poszczególnym częstotliwościom dźwięków wykorzystywanych w muzyce (ich zakres jest ograniczony od 0 do 127). W kolumnie piątej znajdują się wartości głośności dźwięków zawarte w przedziale od 0 (cisza) do 127 (najgłośniejszy). Kolumny szósta i siódma zawierają wartości przedstawione w kolumnach pierwszej i drugiej wyrażone w sekundach.

Macierz nut przedstawia zatem reprezentację pewnej sekwencji komend *MIDI*, która w przyjętym podejściu analizuje jedynie ograniczony zbiór komunikatów *MIDI* przechowywanych w postaci pliku *MIDI*.

Wzorzec muzyczny W jest zdefiniowany jako macierz nut MN :

$$W = MN_{m \times 7}, m = \begin{cases} \frac{l_W}{dur_W} & \text{jeżeli } \frac{l_W}{dur_W} \in N \\ \left\lfloor \frac{l_W}{dur_W} \right\rfloor & \text{jeżeli } \frac{l_W}{dur_W} \notin N \end{cases} \quad (1)$$

gdzie:

- m_{MN} oznacza liczbę wierszy macierzy nut (MN),
- l_W oznacza zadeklarowaną długość wzorca w jednostkach bpm,
- dur_W jest wartością długości dla wszystkich dźwięków zawartych w macierzy MN .

Mini-generator (oznaczony jako Mg) jest podstawowym komponentem w opisywanej koncepcji autorskiego systemu algorytmicznego komponowania muzyki na podstawie wzorców muzycznych. Definicja mini-generatora Mg została przedstawiona poniżej jako:

$$Mg = \{K, MM, WPN, dur_W, l_W, t\}, \quad (2)$$

gdzie:

- K oznacza klasę mini-generatora,
- MM - motyw muzyczny na bazie którego generowany jest wzorzec muzyczny W ,
- WPN - wektor pozycji (wierszy macierzy MN) nut, w których mają być rozmieszczone dźwięki z motywu MM ,
- dur_W - wartość długości dla wszystkich dźwięków generowanego wzorca W określona w jednostkach bpm,
- l_W - długość generowanego wzorca W określona w jednostkach bpm,
- t - tempo generowanego wzorca W określone w jednostkach bpm.

Występujące w wyrażeniu (2) oznaczenie K definiuje klasę do jakiej należy mini-generator Mg . Przynależność do danej klasy determinuje sposób wykorzystania mini-generatora Mg w procesie generowania wzorca muzycznego. W prezentowanym systemie zdefiniowane zostały następujące klasy mini-generatorów:

- generator monofoniczny,
- generator polifoniczny,

— generator perkusyjny.

Generator monofoniczny wykorzystywany jest do generowania monofonicznych fragmentów muzycznych takich jak np. melodia główna czy monofoniczna linia basowa. Generator polifoniczny generuje polifoniczne fragmenty muzyczne takie jak akord muzyczny. Generator perkusyjny to szczególna klasa mini-generatora, którego generowane sekwencje muzyczne (monofoniczne i polifoniczne) są interpretowane jako sekwencje perkusyjne.

Mini-generator Mg do generowania muzyki wykorzystuje strukturę, która stanowi kombinację motywu muzycznego wcześniej zdefiniowanego przez użytkownika. W podstawowej wersji prezentowanego systemu motyw stanowi wektor liczb naturalnych zawartych w przedziale od 0 do 127, traktowanych jako wysokość dźwięku w standardzie *MIDI*. Zbiór możliwych kombinacji zdefiniowanego motywu muzycznego stanowi macierz KMM . Macierz KMM składa się z k -elementowych kombinacji C n -elementowego motywu muzycznego MM , rozłożonych w macierzy o wymiarach m wierszy (liczba kombinacji) na k kolumn (liczba elementów motywu muzycznego)

$$KMM = \begin{bmatrix} {}_1C_k^n \\ {}_2C_k^n \\ \vdots \\ {}_mC_k^n \end{bmatrix}, k \leq n. \quad (3)$$

Z macierzy KMM losowana jest dla mini-generatora Mg kombinacja motywu muzycznego zdefiniowana jako MM . Mini-generator Mg może również przyjmować MM w postaci stałej zdefiniowanej liczby naturalnej. Tak sparametryzowany Mg generuje fragmenty muzyczne, w których wszystkie dźwięki posiadają tę samą wysokość równą stałej MM . W zależności od klasy K oraz rodzaju MM (wektor lub skalar) zdefiniowane zostały różne reguły generowania wzorców.

Za rozmieszczenie dźwięków muzycznych w odpowiednich miejscach macierzy nut generowanej przez mini-generator odpowiada wektor pozycji dźwięków WPN . Struktura wektora WPN jest generowana na podstawie wyrażenia:

$$WPN = \{n : rs : l\}, n, rs < l. \quad (4)$$

Parametr n jest dowolną liczbą naturalną mniejszą od zdefiniowanej liczby wierszy l macierzy nut. Stała rs jest liczbą naturalną określającą krok wypełnienia wektora WPN . Dla przykładu: mając zdefiniowaną liczbę wierszy l (dźwięków) w macierzy jako 32, parametr n ustawiony na wartość 1 oraz stałą rs ustawioną na 4, wektor WPN będzie miał postać $WPN = \{1, 5, 9, 13, 17, 21, 25, 29\}$, co oznacza, że tylko dla zdefiniowanych pozycji WPN macierzy nut mini-generatora Mg może osadzać dźwięki muzyczne podczas tworzenia wzorca muzycznego. Rozmieszczenie dźwięków w macierzy wpływa na rytmikę wzorca muzycznego generowanego przez mini-generator i wykorzystanego następnie do generowania wyjściowej kompozycji.

Zdaniem autora najciekawsze efekty pracy omawianego systemu algorytmicznego komponowania muzyki uzyskuje się poprzez zmiany, a właściwie dopasowanie wszystkich wartości parametrów systemu. Aby potwierdzić swoją tezę autor zaprezentował kilka przykładowych wyników wygenerowanych na podstawie różnych konfiguracji ustawień. Wszystkie kompozycje znajdują się na płycie CD dołączonej do dysertacji.

Omawianym dalej przykładem jest kompozycja *s1 – 415.flac*, do syntezy której wykorzystano 6 kanałów *MIDI* z jedynie 2 instrumentami banku *Evanessence2*, tj. *Ukulele* oraz *Latin(Loops)*. W procesie komponowania autor zdefiniował następujące wartości parametrów systemu:

- *dur* = 0.25 - długość dźwięku w macierzy *NM* wyrażona w sekundach,
- *pattern_l* = 32 - liczba wierszy (dźwięków) macierzy *NM*,
- *n_pattern* = 128 - liczba wzorców muzycznych wygenerowanych przez system,
- *u_pattern* = 64 - liczba wzorców muzycznych wykorzystanych w kompozycji,
- *rt* = [4] - tablica *rt* powtórzeń.

Bazowym motywem dla wzorców muzycznych jest wektor wysokości dźwięków zdefiniowany w postaci: $p = \langle 50, 48, 50, 53, 55, 43, 41, 43, 36, 38 \rangle$, który posłużył do budowy macierzy *KMM*. Macierz *KMM* składa się z trzech “części” zawierających kombinacje wygenerowane na podstawie trzech wersji wektora *p*: podstawowej, zwiększonej o wartość 4 (linia 10) oraz zwiększonej o wartość 7. Tak więc górna “część” macierzy *KMM* zbudowana jest na podstawie kombinacji wektora *p*, środkowa stanowi kombinacje wektora $p_1 = \langle 54, 52, 54, 57, 59, 47, 45, 47, 40, 42 \rangle$ a dolna - kombinacje wektora $p_2 = \langle 57, 55, 57, 60, 62, 50, 48, 50, 43, 45 \rangle$.

Konfiguracja mini-generatorów wykorzystanych w procesie komponowania została przedstawiona w tab. 5. Mini-generatory od *Mg₁* do *Mg₁₁* należą do klasy generującej partie instrumentów perkusyjnych, natomiast mini-generatory od *Mg₁₂* do *Mg₁₈* generują partie melodii, przy czym w tym wypadku są to partie odtwarzane przez jeden instrument muzyczny. Dla urozmaicenia części rytmicznej kompozycji, autor zdefiniował dynamiczną strukturę motywów *PM* dla wzorców *Mg₉*, *Mg₁₀* i *Mg₁₁*.

Dla mini-generatora *Mg₉* motyw *PM* przyjmuje postać wektora *pm*, zbudowanego na podstawie implementacji:

```
step = fix(pattern_l / fix(rand(1)*rhythmic_step(7)+1));
vec = rhythmic_value{7}(1:step:end);
temppattern(vec,4) = [38:1:38+length(vec)-1];
```

Podobnie dla mini-generatora *Mg₁₀* motyw *PM* przyjmuje postać wektora $\widetilde{pm}$, zbudowanego na podstawie implementacji:

```
step = fix(pattern_l / fix(rand(1)*rhythmic_step(7)+1));
vec = rhythmic_value{7}(1:step:end);
temppattern(vec,4) = [35:1:35+length(vec)-1];
```

Mini-generator *Mg₁₁* generuje dźwięki na podstawie wektora $\widehat{pm}$, zdefiniowanego wg. implementacji:

```
step = fix(pattern_l / fix(rand(1)*rhythmic_step(7)+1));
vec = rhythmic_value{1}(1:step:end);
temppattern(vec,4) = ones(1,length(vec))*38;
```

Dla tak zdefiniowanych mini-generatorów utworzona została architektura generowania kompozycji wg schematu przedstawionego na rys. 1. Bloki oznaczone przez *Mg_x* symbolizują zdefiniowane mini-generatory biorące udział w procesie generowania wzorca muzycznego. Od każdego bloku odchodzi pewna liczba odcinków

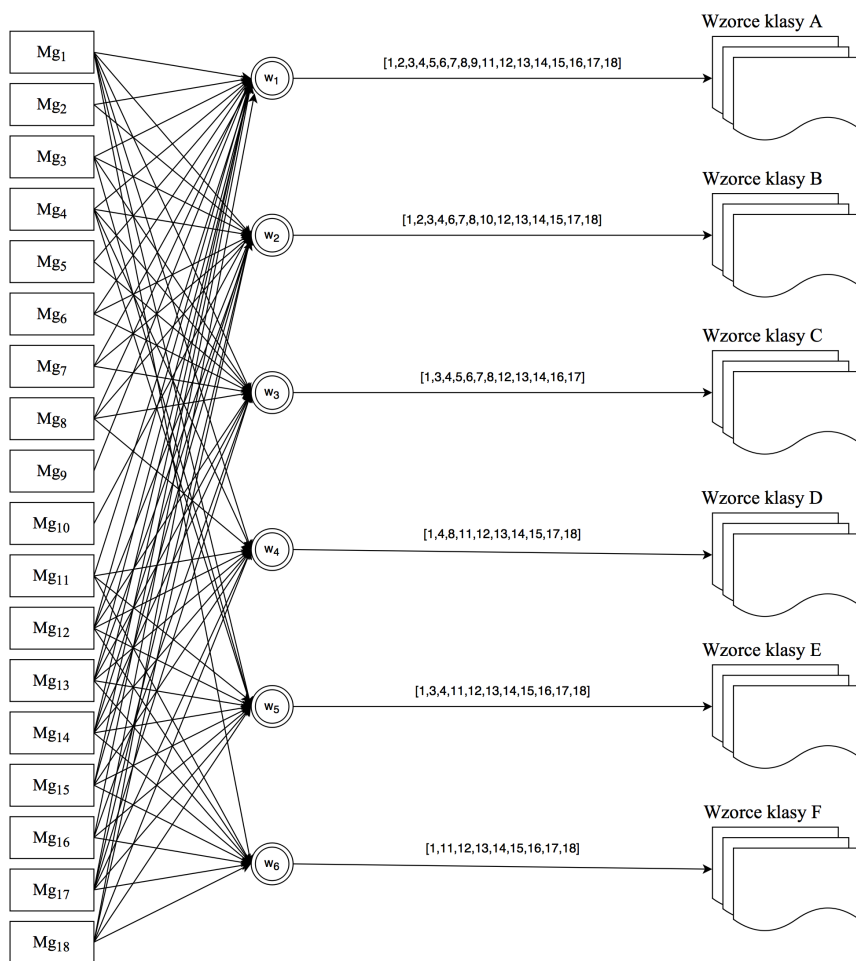
Mini-generator	Klasa	<i>rhythmic_step</i>	<i>rhythmic_value</i>	PM	Kanał <i>MIDI</i>
Mg_1	D	32	[1,5,9,...,29]	36	10
Mg_2	D	32	[2,5,8,...,32]	36	10
Mg_3	D	16	[2,5,8,...,32]	40	10
Mg_4	D	32	[1,2,...,32]	42	10
Mg_5	D	32	[2,5,8,...,32]	51	10
Mg_6	D	16	[16,20,24,28,32]	47	10
Mg_7	D	16	[16,19,22,...,31]	48	10
Mg_8	D	16	[1,5,9,...,29]	54	10
Mg_9	D	8	[17,21,25,29]	pm	10
Mg_{10}	D	8	[17,21,25,29]	$\widetilde{pm}$	10
Mg_{11}	D	8	[1,5,9,...,29]	$\widehat{pm}$	10
Mg_{12}	M	32	[2,6,10,...,30]	z KMM	1
Mg_{13}	M	8	[2,6,10,...,30]	z Mg_{12}	2
Mg_{14}	M	32	[1,6,11,...,31]	z Mg_{12}	3
Mg_{15}	M	16	[2,4,6,...,32]	z Mg_{12}	4
Mg_{16}	M	32	[3,6,9,...,30]	z Mg_{12}	5
Mg_{17}	M	32	[1,2,3,...,32]	z Mg_{12}	6
Mg_{18}	M	16	[2,5,8,...,32]	z Mg_{12}	7

Tab. 5: Parametry zdefiniowanych mini-generatorów dla kompozycji *s1 – 415.flac*. Opracowanie własne.

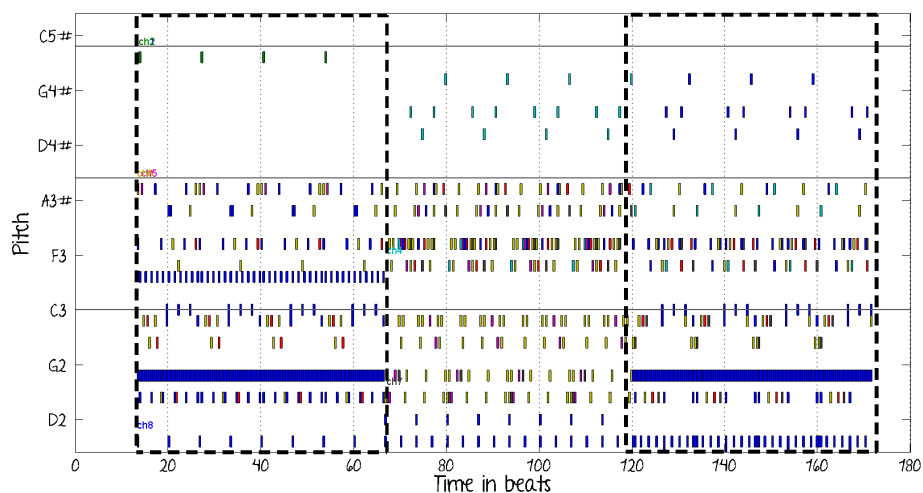
łączących się w jednym z sześciu węzłów *w* symbolizujących proces konkatencji wzorca muzycznego.

Wygenerowane wzorce muzyczne posiadają strukturę złożoną z maksymalnie 32 dźwięków muzycznych, natomiast w kompozycji końcowej występują w postaci grup zawierających cztery powtórzenia wzorca. Właściwość tę widać szczególnie na rys. 2 przedstawiającym fragment kompozycji *s1 – 415.flac* zbudowany z trzech wzorców. Podobieństwo w budowie jest zauważalne przede wszystkim pomiędzy pierwszą a ostatnią częścią przedstawianego fragmentu, w których mini-generatory partii perkusyjnych utworzyły zbliżone struktury rytmiczne (oznaczone na rys. 2).

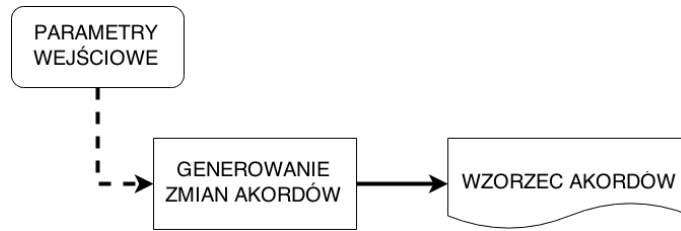
Podobieństwa w budowie poszczególnych grup wzorców są słyszalne również w kolejnych częściach przedstawianej kompozycji. Dodatkowo występujące po sobie kolejne fragmenty oscylują wokół podobnego motywu muzycznego (zbudowanego na podstawie wektora p , p_1 lub p_2). Utworzona kompozycja spełnia definicję muzyki tła (podanej we wcześniejszej części pracy) jednak, w opinii autora, jej “powtarzalność” i swego rodzaju “chaotyczność” mogą stanowić o słabych cechach prezentowanej koncepcji systemu. Podobne cechy komponowanej muzyki można usłyszeć w innych kompozycjach, umieszczonych na dołączonej do pracy płycie CD. Biorąc pod uwagę powyższe stwierdzenia oraz analizując wyniki przeprowadzonych eksperymentów, autor zaproponował inną, odmienną koncepcję algorytmicznego komponowania muzyki, uwzględniającą definicję formy generowanej kompozycji.



Rys. 1: Schemat generowania architektury klas wzorców przez zdefiniowane mini-generatory Mg dla kompozycji $s1 - 415.flac$. Opracowanie własne.



Rys. 2: Wykres pianolowy fragmentu kompozycji $s1 - 415.flac$ z zaznaczonymi grupami wzorców wykazującymi podobieństwo budowy. Opracowanie własne.



Rys. 3: Generowanie zmian akordów. Opracowanie własne.

11.2. Komponowanie uwzględniające opis formy generowanego utworu

W poprzednim rozdziale poświęconym przeprowadzonym badaniom nad użytecznością algorytmicznego systemu komponowania muzyki na podstawie wzorców, autor wyeksponował rolę mini-generatora, jako podstawowej funkcjonalnej części koncepcji tworzenia muzyki bez określania jednoznacznej formy wynikowej kompozycji. Tak wykreowana sztuczna twórczość posiada znamiona muzyki tła, w której dopatrzeć się można teoretycznie nieskończonej struktury, ograniczonej wartościami parametrów systemu. To nieograniczenie formy lokuje omawiany system w dziedzinie systemów komponowania raczej prymitywnych, w których zaszyta muzyczna “wiedza teoretyczna” jest minimalna. Z tego względu autor podjął próbę poszerzenia opisywanej wcześniej koncepcji komponowania o możliwość definiowania finalnej struktury formy kompozycji i jej elementów.

Najważniejsza zmiana dotyczy sposobu generowania wzorców, z których tworzona jest wyjściowa kompozycja muzyczna. Dodatkowo autor zrezygnował z etapu oceny podobieństwa wygenerowanych wzorców, z racji przyjętego mechanizmu generowania i łączenia wzorców w ramach kompozycji.

Biles w swoich rozważaniach na temat modelowania procesu komponowania muzyki poprzez zdefiniowanie elementarnych zadań, które kształtują ten proces wspominał, że rozmieszczenie zadań ulokowanych w tab. 3 zostało wykonane w sposób arbitralny. System, który w sposób mniej lub bardziej kompleksowy generuje kompozycje muzyczne, może bazować na różnych zadaniach procesu komponowania, nie koniecznie w kolejności przedstawionej w tab. 3. Z tego względu autor w tej części pracy zaproponował odmienne podejście do problemu algorytmizacji procesu komponowania muzyki, bazując na harmonizacji jako wejściowym etapie generowania wzorców.

Na rys. 3 zaprezentowano w sposób uproszczony proces powstawania wzorca zawierającego wygenerowaną sekwencję akordów

$$W_A = \{A_p, n_A, l_A, S, R_z\}. \quad (5)$$

Do parametrów wejściowych (rys. 3) kształtujących proces generowania wzorca sekwencji akordów W_A zaliczone zostały:

- A_p - wygenerowany akord początkowy,
- n_A - liczba akordów, które mają być wygenerowane w sekwencji,
- l_A - długość akordu (w jednostkach bpm),
- S - skala muzyczna, na podstawie której generowana będzie sekwencja,
- R_z - zbiór reguł tworzenia zmian (następczości) generowanych akordów.

Sekwencja akordów (stopnie)	Sekwencja akordów (nazwy)
I → IV (kadencja plagalna)	C-dur → F-dur
I → V (kadencja doskonała)	C-dur → G-dur
I → IV → V (triada harmoniczna)	C-dur → F-dur → G-dur
I → IV → V → IV	C-dur → F-dur → G-dur → F-dur
I → V → vi → IV	C-dur → G-dur → a-mol → F-dur
I → ii → IV → V	C-dur → d-mol → F-dur → G-dur
I → ii → IV	C-dur → d-mol → F-dur
I → VI → ii → V	C-dur → A-dur → d-mol → G-dur
I → VI → IV → V	C-dur → A-dur → F-dur → G-dur
IV → I → IV → V	F-dur → C-dur → F-dur → G-dur

Tab. 6: Przykładowe kadencje oparte na tonacji C-dur. Opracowanie własne.

Akord początkowy A_p jest macierzą nut MN zdefiniowaną jako

$$MN = \begin{bmatrix} p_1 m \\ \vdots \\ p_i n \end{bmatrix}, \quad (6)$$

przy czym p określa pojedynczy sparametryzowany dźwięk muzyczny w macierzy MN , natomiast m i n oznaczają stopnie wykorzystanej skali muzycznej (wtedy $m, n \leq 7$) oraz i wyznacza liczbę dźwięków wykorzystanych ($i \geq 3$) w początkowym akordzie A_p .

Parametr n_A określa liczbę kolejnych akordów A wygenerowanych w ramach wzorca W_A . Długość akordu l_A wyraża rozciągnięcie w czasie pojedynczego akordu A w jednostkach bpm.

Szerszego wyjaśnienia wymaga przedstawiony we wzorze (5) zbiór reguł R_z , na podstawie którego budowana jest sekwencja akordów W_A . Zasady generowania sekwencji zostały zapisane w postaci reguł bazujących na kadencjach, które wykorzystują pojęcie *funkcji harmonicznej* budującej lub wygaszającej napięcie w utworze muzycznym.

W tab. 6 przedstawione zostały przykładowe kadencje zbudowane w tonacji C-dur. W kolumnie “Sekwencja akordów (stopnie)” znajdują się sekwencje akordów oznaczonych za pomocą cyfr rzymskich, które odpowiadają stopniom skali. Kolumna “Sekwencja akordów (nazwy)” zawiera przykłady sekwencji akordów dla tonacji C-dur.

Akordy zbudowane na kolejnych stopniach skali w systemie dur-moll tworzą funkcje harmoniczne, wśród których najczęściej występującymi są: *tonika*, *subdominanta* i *dominanta*. Obok wymienionych funkcji w systemie dur-moll występują także tzw. funkcje poboczne, czyli funkcje zbudowane na stopniach skali innych niż I, IV i V. Do tej grupy funkcji należą *medianta* i *submedianta* oraz akordy poboczne zbudowane na II i VII stopniu skali.

Zbiór reguł R_z ujęty we wzorze 5 został przez autora zdefiniowany jako macierz tranzycji (7), w której następstwo pojawienia się określonego akordu j (funkcji harmonicznej) uwarunkowane jest zdefiniowanym prawdopodobieństwem p_{ij} oraz aktualnie wygenerowanym akordem i . W nagłówkach kolumn i wierszy macierzy tranzycji znajdują się cyfry rzymskie, określające funkcje harmoniczne przedstawione w tab.

6. Dla każdego przejścia czyli wygenerowania akordu j będącego następnym w sekwencji po akordzie i , definiowane jest prawdopodobieństwo p_{ij} wystąpienia takiego zdarzenia. W macierzy celowo nie zostały podane konkretne wartości prawdopodobieństwa, gdyż zmieniają się one w zależności od aktualnego stopnia kadencji oraz numeru kolejnego wygenerowanego akordu w ramach progresji. Dla przykładu: prawdopodobieństwo przejścia ze stopnia I do IV stopnia skali w sytuacji gdy akord ten jest jedynym w progresji (początek procesu generowania) jest inne niż w przypadku gdy akord ten został powtórzony w kolejnym kroku (w progresji akordów znajdują się dwa akordy zbudowane na I stopniu skali).

	I	II	III	IV	V	VI	VII
I	p_{11}	p_{12}	p_{13}	p_{14}	p_{15}	p_{16}	p_{17}
II	p_{21}	p_{22}	p_{23}	p_{24}	p_{25}	p_{26}	p_{27}
III	p_{31}	p_{32}	p_{33}	p_{34}	p_{35}	p_{36}	p_{37}
IV	p_{41}	p_{42}	p_{43}	p_{44}	p_{45}	p_{46}	p_{47}
V	p_{51}	p_{52}	p_{53}	p_{54}	p_{55}	p_{56}	p_{57}
VI	p_{61}	p_{62}	p_{63}	p_{64}	p_{65}	p_{66}	p_{67}
VII	p_{71}	p_{72}	p_{73}	p_{74}	p_{75}	p_{76}	p_{77}

Tab. 7: Macierz tranzykcji generowania sekwencji akordów. Opracowanie własne.

Budowa kadencji bazująca na przyjętej koncepcji macierzy tranzykcji sekwencji akordów pozwala definiować zarówno typowe konfiguracje sekwencji (np. triada harmoniczna: tonika $\rightarrow$ subdominanta $\rightarrow$ dominanta), jak i sekwencje niestandardowe (np. funkcje poboczne oraz tzw. wtrącenia, czyli dodanie dominanty do akordu z istniejącej sekwencji).

Kolejnym krokiem w etapie generowania wzorca wykorzystywanego w przyjętej koncepcji jest konkatencja wzorca W_A zdefiniowanego we wzorze 5 ze wzorcem melodii W_M wygenerowanej na podstawie sekwencji akordów W_A . Wzorec melodii W_M jest macierzą nut wygenerowaną na podstawie parametrów przedstawionych w postaci

$$W_M = \{W_A, \{d_1, d_2, \dots, d_n\}, \{o_1, o_2, \dots, o_m\}, \{v_1, v_2, \dots, v_s\}\}, \quad (7)$$

gdzie:

- W_A - wzorec sekwencji akordów, na podstawie którego wygenerowana będzie melodia,
- $\{d_1, d_2, \dots, d_n\}$ - wektor możliwych długości dźwięków występujących w melodii,
- $\{o_1, o_2, \dots, o_m\}$ - wektor możliwych przesunięć dźwięków występujących w melodii,
- $\{v_1, v_2, \dots, v_s\}$ - wektor możliwych głośności dźwięków występujących w melodii.

Algorytm 1 przedstawia koncepcję generowania melodii bazującą na analizie wzorca W_A sekwencji akordów. Mocnym ograniczeniem narzuconym przez autora jest wykorzystanie podczas tworzenia melodii wyłącznie dźwięków wchodzących w skład analizowanych akordów. Głównym elementem algorytmu 1 jest okienkowanie sygnału (wzorca W_A) oknem O_A o zadanej rozpiętości czasowej $< t_p, t_k >$, które zawiera wydzielony fragment wzorca. Podczas przesuwania okna wzdłuż wzorca W_A generowane są kolejne dźwięki D_M melodii, która ostatecznie przybiera postać macierzy nut wzorca W_M . Jest to podejście częściowo stochastyczne, częściowo deter-

Algorytm 1 Generowanie melodii na podstawie sekwencji akordów. Opracowanie własne.

1. Oblicz długość l_A wzorca W_A ,
 2. Zdefiniuj pustą macierz nut dla wzorca melodii W_M i linii basowej W_B ,
 3. Ustaw indeks przeszukiwania i wzorca W_A na początek analizowanego fragmentu: $i = 0$,
 4. Zdefiniuj okno analizy wzorca O_A : początek okna: $t_p \leftarrow i$, koniec okna: $t_k \leftarrow i + \Delta t$,
 5. Dopóki $i + \Delta t \leq l_A$:
 - a) Wybierz z okna analizy wzorca zbiór występujących we fragmencie wysokości dźwięków $\{h_1, h_2, \dots, h_k\}$,
 - b) Ustaw indeks przeszukiwania j okna O_A na: $j \leftarrow t_p$,
 - c) Ustaw indeks położenia generowanego dźwięku melodii $r_{D_M} \leftarrow t_p$,
 - d) Dopóki $j \neq t_k$:
 - i. Dopóki $r_{D_M} \leq t_k$:
 - A. Wylosuj z $\{h_1, h_2, \dots, h_k\}$ wysokość dźwięku h ,
 - B. Wylosuj z $\{d_1, d_2, \dots, d_n\}$ długość dźwięku d ,
 - C. Wylosuj z $\{o_1, o_2, \dots, o_m\}$ przesunięcie dźwięku o ,
 - D. Wylosuj z $\{v_1, v_2, \dots, v_s\}$ głośność dźwięku v ,
 - E. Utwórz dźwięk muzyczny D_M o wysokości h , długości d , głośności v oraz definiowanym przesunięciem w czasie o ,
 - F. Dodaj dźwięk muzyczny D_M do macierzy nut wzorca melodii W_M : $W_M \leftarrow \{W_M; D_M\}$,
 - ii. Zwiększ indeks r_{D_M} o wartość położenia dźwięku muzycznego o : $r_{D_M} \leftarrow r_{D_M} + o$,
 - e) Generuj wzorzec linii basowej W_B na podstawie zawartości wzorca W_M i aktualnego położenia okna O_A ,
 - f) Zwiększ indeks przeszukiwania j : $j \leftarrow j + i$,
 - g) Przesuń indeks przeszukiwania i : $i \leftarrow i + \Delta t$,
 - h) Przesuń okno analizy wzorca O_A : $t_p \leftarrow i$, $t_k \leftarrow i + \Delta t$,
 6. Zwróć macierz nut wzorca W_M .
-

ministyczne, ponieważ dźwięki melodii losowane są ze zbioru możliwych dźwięków występujących w akordach, natomiast ich rozłożenie w czasie jest z góry ograniczone występowaniem akordów w macierzy wzorca W_A .

Uzupełnieniem nowej koncepcji generowania wzorca muzycznego jest wprowadzenie mechanizmu tworzenia wzorca linii basowej (oznaczonego jako W_B) na podstawie generowanego wzorca W_M . Wymienioną funkcjonalność, o której wspomniano w kroku 5e w algorytmie 1, przedstawiono szerzej w algorytmie 2.

Generowanie linii basowej polega na fragmentacji linii melodycznej wzorca W_M oraz transponowaniu o m oktawy w dół. Ten prosty zabieg powoduje poszerzenie spektrum częstotliwości dźwięków muzycznych w wygenerowanej kompozycji, a linia basowa jest ściśle związana z melodią główną. Należy przy tym dodać, że dla każdego okna analizy proces generowania linii basowej rozpoczyna się na nowo, co w konsekwencji powoduje urozmaicenie rytmiczne występujących we wzorcu W_B dźwięków.

W ramach przedstawionego procesu generowania wzorców autor założył również

Algorytm 2 Generowanie linii basowej na podstawie melodii. Opracowanie własne.

1. Wybierz fragment wzorca melodii W_M zgodnie z położeniem okna analizy O_A ,
 2. Zdefiniuj parametr k jako liczbę losową z przedziału $< 1, \dots, l >$, gdzie l stanowi liczbę wszystkich dźwięków muzycznych w wybranym fragmencie wzorca W_M ,
 3. Z wybranego fragmentu macierzy nut wzorca melodii wybierz n dźwięków, pomijając co k -ty element,
 4. Transponuj o m oktawy w dół dźwięki wybrane w punkcie 3,
 5. Dodaj transponowane dźwięki do macierzy wzorca W_B linii basowej.
-

możliwość stosowania zdefiniowanego wcześniej (2) mini-generatora. Należy jednak przedstawić zasadniczą różnicę pomiędzy wykorzystaniem mini-generatora w koncepcji generowania utworu muzycznego, w którym nie została zdefiniowana forma kompozycji, a koncepcją, w której opis formy utworu stanowi centralny punkt procesu komponowania.

W poprzednim rozdziale mini-generatory zostały przedstawione jako mechanizm generowania wszelkich wzorców muzycznych, wykorzystywanych później w procesie generowania kompozycji wyjściowej. Koncepcja komponowania utworu z uwzględnieniem definiowania jego formy wykorzystuje mini-generatory, w których parametr MM przedstawiony we wzorze (2) przyjmuje postać skalarą - wartości odpowiadającej konkretnemu instrumentowi perkusyjnemu zdefiniowanemu w standardzie General Midi. Dodatkowo parametr K oznaczający klasę mini-generatora jest wspólny dla wszystkich mini-generatorów wykorzystanych podczas generowania wzorca i oznaczony jako klasa generatora perkusyjnego.

Istotną zmianą w stosunku do przedstawionej w poprzednim rozdziale koncepcji generowania kompozycji wyjściowej jest pominięcie etapu selekcji wzorców i generowanie utworu muzycznego na podstawie modelu kompozycji $\hat{U}$

$$\hat{U} = \{PW, F = \{i_{f_1}; i_{f_2}; \dots; i_{f_n}\}, S, I, \{MG_{f_1}; MG_{f_2}; \dots; MG_{f_n}\}\}, \quad (8)$$

w którym:

- PW - oznacza zbiór parametrów wejściowych kompozycji,
- $F = \{i_{f_1}; i_{f_2}; \dots; i_{f_n}\}$ - oznacza zbiór identyfikatorów zdefiniowanych fragmentów muzycznych $f_1 \dots f_n$ opisujący ich rozmieszczenie w kompozycji,
- S - oznacza zbiór kanałów *MIDI* wykorzystanych w kompozycji,
- I - oznacza zbiór instrumentów *MIDI* przypisanych do zdefiniowanych w kompozycji kanałów *MIDI*,
- $\{MG_{f_1}; MG_{f_2}; \dots; MG_{f_n}\}$ - oznacza zbiór mini-generatorów MG przypisanych do poszczególnych fragmentów kompozycji $f_1 \dots f_n$, przy czym n stanowi liczbę zdefiniowanych fragmentów kompozycji.

Do parametrów wejściowych PW kształtujących kompozycję wyjściową systemu należą:

- tempo utworu (liczba całkowita, wyrażająca szybkość odtwarzania kompozycji w jednostkach bpm),
- przyjęta skala muzyczna, w której tworzona będzie kompozycja,
- definicje fragmentów muzycznych, z których będzie generowana kompozycja,
- długości i liczby akordów wchodzących w skład zdefiniowanych fragmentów kompozycji.

Skala muzyczna została zdefiniowana jako wektor 8-mio elementowy, w którym znajdują się wartości interwałów kolejnych stopni skali, wyrażone w liczbie półtonów. Pierwszym elementem tego wektora jest liczba 0, która opisuje pierwszy stopień skali - wartość odniesienia dla pozostałych stopni. Ostatni element wektora został zdefiniowany jako interwał oktawy wyrażony wartością 12 półtonów. Przykładowo dla naturalnej skali durowej zdefiniować można następujący wektor interwałów: $\langle 0, 2, 4, 5, 7, 9, 11, 12 \rangle$. W takim wypadku dla dźwięku $C4$, dla którego wysokość wyrażona wg standardu *MIDI* wynosi 60, otrzymujemy naturalną skalę C-dur w następujący sposób:

$60 + \langle 0, 2, 4, 5, 7, 9, 11, 12 \rangle = \langle 60, 62, 64, 65, 67, 69, 71, 72 \rangle$, przy czym ostatnim elementem tak utworzonego wektora jest dźwięk $C5$, należący już do kolejnej oktawy.

Fragment muzyczny stanowi zdefiniowaną część kompozycji przedstawioną w postaci macierzy nut zbudowanej na podstawie wybranych wzorców muzycznych. Formalny zapis fragmentu jest zdefiniowany jako

$$f = \{i_f, S_f, I_f, W_{A_f}, W_{M_f}, W_{B_f}, MG_f\}, \quad (9)$$

gdzie:

- i_f jest identyfikatorem danego fragmentu f ,
- S_f jest zbiorem kanałów *midi* wykorzystanych do wygenerowania fragmentu f ,
- I_f stanowi zbiór instrumentów *MIDI* przypisanych do zdefiniowanych we fragmencie f kanałów *MIDI*,
- W_{A_f} jest wzorcem sekwencji akordów przypisanym do fragmentu f ,
- W_{M_f} jest wzorcem melodii wygenerowanej na podstawie W_{A_f} i przypisanym do fragmentu f ,
- W_{B_f} jest wzorcem linii basowej wygenerowanej na podstawie W_{M_f} i przypisanym do fragmentu f ,
- MG_f stanowi zbiór mini-generatorów przypisanych do fragmentu f .

Identyfikator i_f fragmentu służy do jego lokalizowania w zbiorze F (8) opisującym formę generowanej przez system kompozycji muzycznej. Przykładowo dla zdefiniowanych fragmentów kompozycji f_1, f_2, f_3 i f_4 można zdefiniować ich identyfikatory: $i_1 = 1, i_2 = 2, i_3 = 3$ oraz $i_4 = 4$. Dla tak zdefiniowanych danych zbiór F może przykładowo przybrać postać $F = [1, 1, 2, 2, 2, 2, 3, 3, 2, 2, 2, 2, 3, 3, 4, 4]$. Zbiór ten kolekcję należy analizować od lewej do prawej strony, przy czym kolejność elementów zbioru jest istotna. W przedstawionym przykładzie zbioru F można zauważyć, że fragment f_1 o identyfikatorze $i_1 = 1$ występuje dwukrotnie na początku kompozycji. Następnie naprzemiennie występują dwie grupy fragmentów: czterokrotnie powtórzony fragment f_2 ($i_2 = 2$) oraz dwukrotnie powtórzony fragment f_3 ($i_3 = 3$). Kompozycja kończy się dwukrotnym wystąpieniem fragmentu f_4 o identyfikatorze $i_4 = 4$.

S_f i I_f (10) stanowią podzbiory występujących kolejno w (8) zbiorów: S kanałów *MIDI* oraz I instrumentów *MIDI* wykorzystanych w kompozycji $\hat{U}$. Na tym etapie badań autor założył występowanie w danym fragmencie jednego wzorca sekwencji akordów W_{A_f} , na którym bazuje mechanizm generowania melodii i linii basowej. Istnieje natomiast możliwość generowania więcej niż jednej melodii i linii basowej dla wygenerowanego wzorca W_{A_f} , co zostało szerzej przedstawione w części dysertacji poświęconej eksperymentom i wynikom badań.

Ostatnimi parametrami ujętymi w zbiorze parametrów wejściowych PW są długości akordów (wyrażone w jednostkach bpm) zdefiniowane niezależnie dla każdego fragmentu kompozycji oraz liczby akordów w ramach wzorców sekwencji akordów W_{A_f} związanych z fragmentami kompozycji. Wymienione parametry służą do określenia długości wzorca sekwencji akordów $l_{W_{A_f}}$ wyrażonej wzorem

$$l_{W_{A_f}} = l_A * n, \quad (10)$$

gdzie:

- l_A - długość akordu (w bpm),
- n - liczba akordów generowanych we wzorcu W_{A_f} .

Długość wzorca sekwencji akordów W_{A_f} jest wykorzystywana podczas generowania melodii i linii basowej (algorytm 2). Odpowiednio dobrane l_A oraz n we wzorcach sekwencji umożliwiają generowanie kompozycji w różnym metrum (l_A) oraz z różnym poziomem złożoności fragmentów kompozycji (n).

Na potrzeby eksperymentów zdefiniowane zostały cztery części kompozycji oznaczone kolejnymi liczbami całkowitymi. Poniżej zostały zdefiniowane następujące wartości parametrów systemu:

- $form = [1, 1, 3, 3, 1, 1, 4, 3, 3, 1]$ - wektor definiujący strukturę kompozycji (kolejność poszczególnych części kompozycji),
- $tempo = 160$ - tempo kompozycji wyrażone w jednostce bpm,
- $scale = [0, 2, 3, 5, 7, 8, 10, 12]$ - wektor opisujący skalę muzyczną, w której wygenerowana zostanie kompozycja (w tym przypadku jest to skala naturalna molowa),
- $possible_location = [0, 8]$ - wektor możliwych lokalizacji dźwięków we wzorcach muzycznych,
- $possible_location_snare = [0, 16]$ - j.w. z tym, że zdefiniowany dla konkretnego instrumentu (werbla),
- $possible_offset = [0]$ - wektor możliwych przesunięć dźwięków w czasie,
- $possible_offset_snare = [8]$ - j.w. z tym, że zdefiniowany dla werbla,
- $offset_mel = [0, 0.5]$ - wektor przesunięć dźwięków zdefiniowany dla melodii,
- $possible_duration_mel1 = [0.25]$ - wektor możliwych długości dźwięku dla melodii,
- $possible_duration_mel2 = [0.5]$ - alternatywny wektor możliwych długości dźwięku dla melodii,
- $possible_duration = [0.25, 0.5]$ - kolejny wektor możliwych długości dźwięku,
- $lp = 4$ - długość akordu wyrażona w jednostce bpm,
- $step_1 = 4$ - liczba akordów w ramach wzorca muzycznego,
- $step_2 = 8$ - alternatywna liczba akordów,
- $fill = [36, 50, 48, 47, 45, 43, 40, 49, 57, 66, 65, 40, 39]$ - pomocniczy wektor wykorzystany przez mini-generator do symulacji przejścia perkusyjnego.

W przedstawionej konfiguracji system generuje cztery rodzaje “fragmentów” kompozycji, przy czym nie wszystkie z nich muszą być wykorzystane w wynikowym utworze. Dodatkowo niektóre wartości parametrów, takie jak wektor długości dźwięku czy możliwych przesunięć, zostały zdefiniowane wielokrotnie w celu wprowadzenia urozmaiconych zbiorów wartości wykorzystanych przez różne mini-generatory. Domyślnym akordem dla generowania progresji akordów jest akord molowy $G_3\#$ (składający się z dźwięków: $G_3\#, B_3, D_4\#$), na podstawie którego budowane są kadencje w poszczególnych częściach kompozycji.

Mini-generator	Klasa	<i>rhythmic_step</i>	<i>rhythmic_value</i>	PM	Kanał <i>MIDI</i>
Mg_1	D	pl	onset	36	10
Mg_2	D	pls	onset_s	40	10
Mg_3	D	pl	onset	42	10
Mg_4	D	pl	onset	51	10
Mg_5	D	pl	onset_h1	pm_h	10
Mg_6	D	pl	onset	49	10
Mg_7	D	pl/2	onset_h1	pm_h	10
Mg_8	D	pl/2	onset_h2	pm_h	10
Mg_9	D	pl	onset	54	10

Tab. 8: Parametry zdefiniowanych mini-generatorów. Opracowanie własne.

W przyjętej koncepcji model mini-generatora odgrywa kluczową rolę podczas budowania wzorca muzycznego dla instrumentów perkusyjnych. W tab. 8 przedstawiona została konfiguracja mini-generatorów zdefiniowanych w badanym systemie komponowania, a dla potrzeb przeprowadzonych eksperymentów autor zdefiniował następujące zmienne przedstawione na listingu poniżej (kod Matlab):

```

lpos = length(possible_location);
lpos_snare = length(possible_location_snare);
lof = length(possible_offset);
lof_snare = length(possible_offset_snare);
...
pl = possible_location(floor(rand(1)*lpos+1));
pls = possible_location_snare(floor(rand(1)*lpos_snare +1));
po = 0;
pos = possible_offset_snare(floor(rand(1)*lof_snare+1));

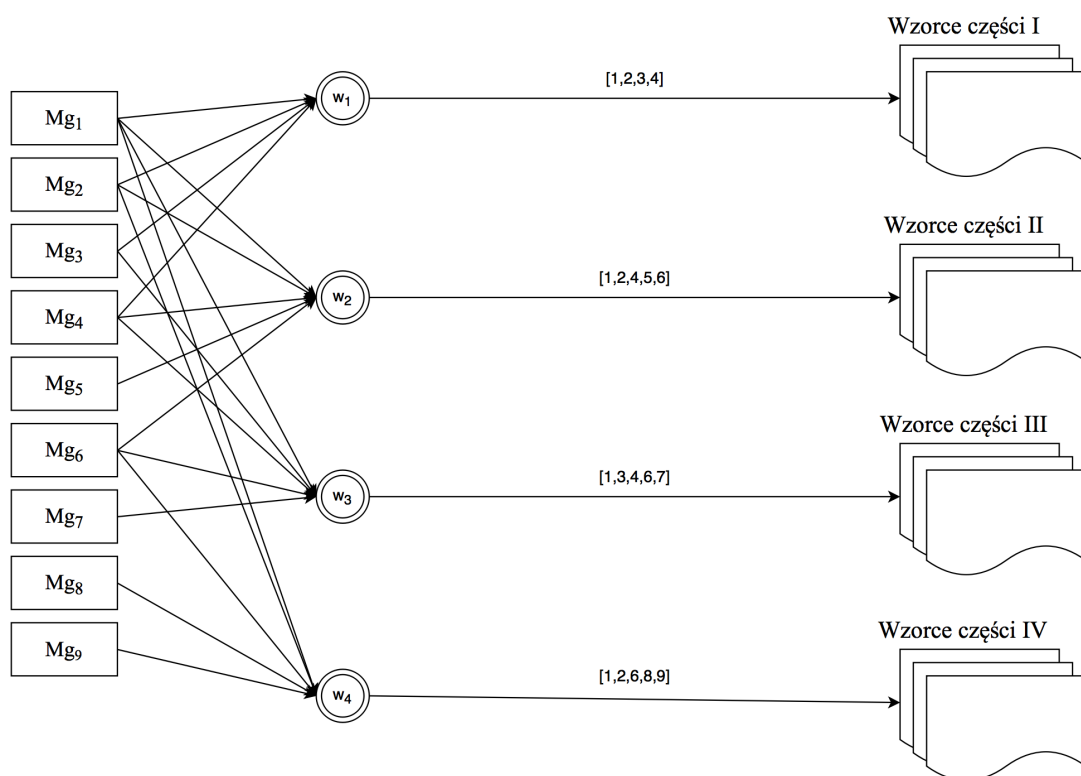
onset = [1+po:pl:size(notematrix,1)];
onset_s = [1+pos:pls:size(notematrix,1)];
onset_h1 = [1+64:pl:size(notematrix,1)];
onset_h2 = [1+po:pl/2:size(notematrix,1)];

fill = [36,50,48,47,45,43,40,49,57,66,65,40,39];
lfill = length(fill);
pm_h = fill(fix(rand(1,8)*lfill)+1);

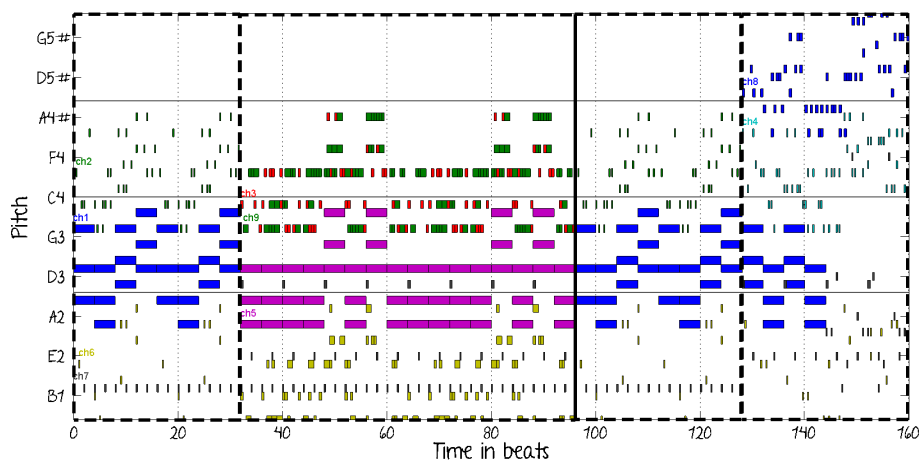
```

Zmienne *pl* i *pls* określają wartości parametru *rhythmic_step* losowane z wektorów lokalizacji dźwięków. Zmienne *onset*, *onset_s*, *onset_h1* i *onset_h2* reprezentują wektory parametru *rhythmic_value* - lokalizacji dźwięków w obrębie wzorca muzycznego. Mini-generatory Mg_5 , Mg_7 i Mg_8 zostały zdefiniowane do budowania ciągów uderzeń w różne instrumenty perkusyjne (podane w konwencji *MIDI* w wektorze *fill*), których strukturę opisuje zmienna *pm_h*.

Wraz z definicją mini-generatorów z tab. 8 zdefiniowana została architektura tworzenia klas wzorców dla poszczególnych części kompozycji muzycznej. Każdą część z osobna można traktować jako odrębną klasę, dla której generowane są wzorce muzyczne wg połączeń mini-generatorów przedstawionych na rys. 4.



Rys. 4: Schemat generowania architektury klas wzorców przez mini-generatory zdefiniowane w tab. 8. Opracowanie własne.



Rys. 5: Fragment kompozycji wygenerowanej na podstawie ustawień ze strony 23. Opracowanie własne.

Fragment przykładowej kompozycji *k_raw_421.flac* zbudowanej na podstawie przedłożonej konfiguracji początkowej zaprezentowano na rys. 5. Wskazany wykres pianolowy został podzielony na obszary, w których występują fragmenty zdefiniowane wg parametru *form*. I tak, w pierwszym od lewej strony obszarze znajdują się dwa powtórzenia fragmentu pierwszego, w kolejnym dwa drugiego, w trzecim następuje ponowne dwukrotne wystąpienie fragmentu pierwszego, natomiast w ostatnim obszarze znajduje się fragment czwarty. Dodatkowo rys. 5 ukazuje różne

długości wygenerowanych wzorców muzycznych: fragment trzeci składa się ze wzorców zbudowanych na podstawie ośmiu wystąpień akordów (parametr *step₂*), podczas gdy fragment pierwszy składa się ze wzorców zawierających cztery akordy (parametr *step₁*).

12. Spis publikacji własnych

1. Mazurowski Ł.: *Generative electronic background music system*, AIP Conf. Proc. 1648, 660004 (2015), <http://dx.doi.org/10.1063/1.4912880>, Rodos, 22-28 września 2014,
2. Mazurowski Ł.: *Algorithmic composition transformational-generative system for background music generation*, Przegląd Elektrotechniczny, R. 91, nr 2, 2015, s. 112-117,
3. Mazurowski Ł.: *Przegląd systemów kształcenia głosu śpiewaczego z wizualnym sprzężeniem zwrotnym*, w *Metody komputerowe w zastosowaniu do analizy i wspierania kształcenia głosu śpiewaczego* (red. E. Półrolniczak), ISBN 978-83-936799-3-5, Stowarzyszenie Przyjaciół Wydziału Informatyki w Szczecinie, 2013, s. 9-24,
4. Mazurowski Ł.: *Pozyskiwanie informacji multimedialnych – bieżące podejścia i zastosowanie*, Firma i Rynek 2013/1 (44), Zeszyty Naukowe Zachodniopomorskiej Szkoły Biznesu w Szczecinie, s. 41-50,
5. Mazurowski Ł., Wiliński A.: *Computer models of similarity in musical pieces based on the multi-dimensional patterns*, Journal of Applied Computer Science, vol. 20, nr 2, 2012, s. 79-94,
6. Mazurowski Ł.: *Computer models for algorithmic music composition*, in Federated Conference on Proceedings of Computer Science and Information Systems (FedCSIS), 2012, IEEE Xplore Digital Library, s. 733-737,
7. Mazurowski Ł.: *An algorithmic composition - the chosen aspects*, Przegląd Elektrotechniczny, Vol 2012, Nr 10b, s. 243-247,
8. Mazurowski Ł.: *Synteza dźwięku w technologii SoundFont*, Modele inżynierii teleinformatyki. Wybrane zastosowania 5, Wydawnictwo Uczelniane Politechniki Koszalińskiej, 2010,
9. Mazurowski Ł.: *Łańcuchy Markowa w systemach generowania muzyki sztucznej*, Modele inżynierii teleinformatyki. Wybrane zastosowania 4, Wydawnictwo Uczelniane Politechniki Koszalińskiej, 2009,
10. Mazurowski Ł., Wiliński A.: *Koncepcja generowania utworu muzycznego przez maszynę - akt sztucznej twórczości opartej na algorytmach genetycznych*, Metody Informatyki Stosowanej 3/2008, PAN, 2008

13. Podsumowanie

W koncepcji algorytmicznego komponowania muzyki na podstawie wzorców z pominięciem formalnego opisu formy generowanej kompozycji autor przeanalizował cztery parametry przyczyniające się do zmiany jej struktury, tj. długość dźwięków muzycznych, długość wzorca, strukturę zbioru wzorców oraz strukturę tablicy powtórzeń wzorców.

Przedstawione wyniki zmian długości dźwięków muzycznych jak i całych wzorców sugerują ich wpływ na powtarzalność fragmentów kompozycji oraz złożoność jej struktury. Przykłady wygenerowanych kompozycji ukazują możliwości systemu w zakresie tworzenia muzyki o zmiennej dynamice i złożoności struktury. Przy przyjętym stałym tempie utworu oraz stałej długości wzorca muzycznego, zmiana parametru długości tworzących go dźwięków powoduje zmianę w dynamice całej kompozycji: im krótsze dźwięki, tym bardziej dynamiczny wydaje się być efekt końcowy pracy systemu. Natomiast w sytuacji gdy przyjęta została stała długość dźwięków muzycznych i tempa kompozycji, zwiększenie długości wzorca powoduje zwiększenie ogólnej złożoności kompozycji - wzorce muzyczne zbudowane są z większej liczby dźwięków, które mogą tworzyć bardziej rozbudowane tematy muzyczne.

Autor w ramach doświadczeń przeprowadził również próby zmian proporcji pomiędzy zbiorem wygenerowanych a wykorzystanych wzorców kompozycji muzycznej oraz zmian budowy tablicy powtórzeń wzorca *rt*. W wyniku przeprowadzenia kilku procesów komponowania otrzymano wynik nacechowany wielokrotnymi powtórzeniami tego samego wzorca, co odsłania pewną słabość opisywanej koncepcji systemu w przypadku słabo urozmaiconego zbioru wzorców muzycznych. Jednocześnie inne kompozycje ukazują różne aspekty powtórzeń wzorców muzycznych w ujęciu kompletnej kompozycji oraz chaotyczności jej struktury w przypadku rozbudowy tablicy powtórzeń *rt*.

Przedstawione wyniki badań nad możliwościami systemu algorytmicznego komponowania muzyki na podstawie wzorców muzycznych i opisu formy utworu, pokazują odmienny charakter tworzonej muzyki, w porównaniu z systemem bazującym wyłącznie na mini-generatorach, a którego analizy dokonano w rozdziale 11.1. Wyodrębnionych 6 parametrów sterujących pracą systemu stanowi jedynie podstawę możliwości modyfikowania końcowej formy kompozycji.

Modyfikacja struktury wektora formy razem ze zdefiniowaniem algorytmu generującego poszczególne części kompozycji to podstawowe narzędzia kreujące finalną postać dzieła muzycznego utworzonego przez proponowany system. Zmiana kolejności elementów formy może być pomocna w eksploracji ciekawych i niesza-blonowych wyników, a sposób definiowania części kompozycji wprowadza szerokie możliwości kreowania dużych fragmentów formy.

Istotnym elementem kształtującym część harmoniczną generowanego utworu jest parametr opisujący skalę muzyczną oraz akord bazowy, na których zbudowane są sekwencje akordów w kompozycji wynikowej. Wynika to ze specyfiki przyjętej koncepcji generowania linii melodycznych oraz linii basowych, które niejako "śledzą" przebieg sekwencji akordu, a budulcem dla nich jest budowa poszczególnych elementów tej progresji. Zatem z pomocą odpowiednio dobranych skali oraz akordu bazowego, można zbudować utwór np. o charakterze smutnym (w tonacjach molo-wych) lub bardziej radosnym (w tonacjach durowych).

Aspekt rytmiczny jest obok harmonii i melodyki jedną z najważniejszych cech generowanego utworu, dlatego też w przyjętej koncepcji zdefiniowane zostały odpowiednie parametry, pozwalające zmieniać jego strukturę rytmiczną. Modyfikacjom rytmicznym mogą podlegać zarówno poszczególne dźwięki (poprzez zmiany parametrów długości i przesunięć dźwięków w obrębie wzorca muzycznego lub części kompozycji) jak i całe fragmenty muzyczne, których charakterystykę można modyfikować za pomocą odpowiednio dobranych wartości liczby i czasu trwania akordów. Zmiana wymienionych parametrów prowadzić może również do zmiany artykulacji

dźwięków, co w połączeniu z ciekawym brzemieniem instrumentu syntezującego daną partię, może dać ciekawe efekty końcowe.

Parametry zdefiniowane w systemie w różnym stopniu pokrywają domeny muzyki związane z procesem komponowania utworu. Autor prezentując wyniki swoich badań miał na celu jedynie przedstawienie określonych technik sterowania tym procesem w obrębie przyjętych koncepcji.

Nawiązując do komponentów problemu badawczego sformułowanych na str. 5 - celu i hipotezy badawczej należy ocenić, że zostały one osiągnięte. Cel opisany jako *opracowanie modelu komputerowego systemu komponowania muzyki użytkowej (posiadającej wartość rynkową), w którym do generowania oryginalnych utworów muzycznych wykorzystuje się metody autorskie oparte na metodach sztucznej inteligencji*, został bez wątpienia osiągnięty. W pracy zostały szczegółowo opisane owe autorskie metody z podkreśleniem ich nowatorskich aspektów. Autor uważa je za własne, pozbawione wad prawnych i innowacyjne wg często przywoływanego dziś kryterium - są nowe i mają wartość (metodyczną, techniczną i, być może, rynkową).

Hipoteza opisana jako przypuszczenie, że *bazując na metodach sztucznej inteligencji, możliwe jest opracowanie takich algorytmów komponowania utworów muzycznych, których efektem będą utwory atrakcyjne w odbiorze i posiadające strukturę poprawną w aspekcie teorii muzyki*, zdaniem autora także została dowiedziona, przy czym owa teoria muzyki stanowi koncept reguł i struktur zdefiniowanych w ramach struktur określających możliwe sąsiedztwa między różnymi zdarzeniami muzycznymi: dźwiękami, całymi akordami czy fragmentami kompozycji.

Literatura

- [1] Trącić myszką. komputerowy kompozytor w domowym studiu nagraniowym. Audycja radiowa, 2011. [dostęp online 04.03.2017: <http://www.myszka.org/archiwum-myszki/tracic-myszka-2011/06112011-komputerowy-kompozytor-w-domowym-studiu-nagraniowym>].
- [2] Trącić myszką. synteza i przetwarzanie dźwięku. Audycja radiowa, 2011. [dostęp online 04.03.2017: <http://www.myszka.org/archiwum-myszki/tracic-myszka-2011/17102011-synteza-i-przetwarzanie-dzwieku>].
- [3] Biles J. A. Genjam: a genetic algorithm for generating jazz solos. *Proceedings of ICMC'94*, pages 131–137, 1994. University of Michigan Library, MI: MPublishing.
- [4] Biles J. A. Genjam: Evolutionary computation gets a gig. In *Third Conference on Information Technology Curriculum*, Rochster, New York, 2002.
- [5] Biles J. A. Genjam in transition: from genetic jammer to generative jammer. In *International Conference on Generative Art*, Milan, Włochy, 2002.
- [6] Gartland-Jones A. Musicblox: A real-time algorithmic composition system incorporating a distributed interactive genetic algorithm. *Applications of Evolutionary Computing: EvoWorkshops 2003. LNCS*, 2611:490–501, 2003.
- [7] Bel B. Migrating musical concepts: an overview of the bol processor. *Computer Music Journal*, 22(2).
- [8] Jacob B. Algorithmic composition as a model of creativity. *Organised Sound*, 1.
- [9] Jacob B. Composing with genetic algorithms. In *Proceedings of the 1995 International Computer Music Conference*, 1995.
- [10] Bentley P. Blackwell T.M. Improvised music with swarms. In *Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on*, volume 2, pages 1462–1467, 2002.
- [11] Ariza C. Prokaryotic groove: Rhythmic cycles as real-value encoded genetic algorithms. In *Proceedings of the 2002 International Computer Music Conference*, 2002.
- [12] Jain S. Wadhwa S. Chaudhury S., Nag T.C. Role of sound stimulation in reprogramming brain connectivity. *Journal of Biosciences*, 38(3):605–614, 2013.
- [13] Miikkulainen R. Chen C. C. J. Creating melodies with evolving recurrent neural network. *Proceedings of IJCNN'01*, 2001.
- [14] Krumhansl C.L. *Cognitive Foundations of Musical Pitch*. Oxford University Press, New York, 1990.
- [15] Dale N. Cockerton T., Moore S. Cognitive test performance and background music. *Perceptual and Motor Skills*, 85:1435–1438, 1997.
- [16] Hofstadter D.R. Cope D. *Virtual music: computer synthesis of musical style*. Mass.: MIT Press, Cambridge,, 2001.
- [17] Conklin D. Music generation from statistical models. In *Proceedings of the AISB 2003 - Symposium on Artificial Intelligence and Creativity in the Arts and Sciences*, pages 30–35, 2003.
- [18] Cope D. *Computers and Musical Style*. A-R Editions, 1991.
- [19] Cope D. Recombinant music using the computer to explore musical style. *IEEE Computer*, 24:22–28, 1991. IEEE Computer Society Press Los Alamitos.
- [20] Cope D. Computer modeling of musical intelligence in emi. *Computer Music Journal*, 16:69–83, 1992. The MIT Press.
- [21] Cope D. *Experiments in musical intelligence*. A-R Editions, 1996.
- [22] Cope D. *The algorithmic composer*. A-R Editions, 2000.
- [23] Ralley D. Genetic algorithm as a tool for melodic developmenty. In *Proceedings of the 1995 International Computer Music Conference*, 1995.
- [24] Collins English Dictionary. background music. Dostęp on-line: 26-07-2014.

- [25] Ames Ch. Domino M. Understanding music with ai. chapter Cybernetic composer: an overview, pages 186–205. MIT Press, Cambridge, MA, USA, 1992.
- [26] Toiviainen P. Eerola T. Mir in matlab: The midi toolbox. *Proceedings of 5th International Conference on Music Information Retrieval*, pages 22–27, 2004.
- [27] Correa D.C. et al. Neural network based systems for computer-aided musical composition: supervised x unsupervised learning. *SAC*, pages 1738–1742, 2008. ACM.
- [28] Dubnov S. et al. Using machine-learning methods for musical style modeling. *IEEE Computer*, 36(10), 2003.
- [29] Lartillot O. et al. Automatic modeling of musical style. *Proceedings of ICMC'02*, 2002.
- [30] Schoner B. Farbood M. Analysis and synthesis of Palestrina-style counterpoint using Markov chains. In *International Computer Music Conference*, 2001.
- [31] Copley P. Gartland-Jones A. The suitability of genetic algorithms for music composition. *Contemporary Music Review*, 22:43–55, 2003.
- [32] Tuson A. Geraint Wiggins G. Papadopoulos G., Phon-amnuaisuk S. Evolutionary methods for musical composition. In *International Journal of Computing Anticipatory Systems*, 1998.
- [33] Slimane M. Guéret C., Monmarché N. Ants can play music. In Blum C.-Gambardella L.M. Mondada F. Stützle T. Dorigo M., Birattari M., editor, *Ant Colony Optimization and Swarm Intelligence*, volume 3172 of *Lecture Notes in Computer Science*, pages 310–317. Springer Berlin Heidelberg, 2004.
- [34] Ayres L. Horner A. Harmonisation of musical progression with genetic algorithms. In *Proceedings of the 1995 International Computer Music Conference*, 1995.
- [35] Info-wiadomości. Festiwal nauki i sztuki oraz dzień otwarty w i liceum ogólnokształcącym im. adama mickiewicza w stargardzie, 2014. [dostęp online 08.03.2017: <http://www.wiadomosci.rii.pl/index.php?H=11&HH=9970>].
- [36] Renkewitz F. Kampfe J., Sedlmeier P. The impact of background music on adult listeners: A meta-analysis. *Psychology of Music*, 39:424–448, 2011.
- [37] University College London. Compgc20 - computer music, 2016. [dostęp online 07.03.2017: http://www.cs.ucl.ac.uk/students/syllabus/msccs/gc20_computer_music/].
- [38] Hoggar S. McAlpine K., Miranda E.R. Making music with algorithms: A case-study system. *Computer Music Journal*, 23(2).
- [39] Jung-Ju C. Min-Joon Y., In-Kwon L. *Background Music Generation Using Music Texture Synthesis*, volume 3166 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2004.
- [40] Biles J. A. Miranda E. R. *Evolutionary Computer Music*. Springer-Verlag, Londyn, 2007.
- [41] Tominaga K. Miura T. An approach to algorithmic music composition with an artificial chemistry. In *Proc. of the German Workshop on Artificial Life*, 2006.
- [42] Reagan P. *Computer music generation via decision tree learning. Master Thesis*. Department of Computer Science, Carnegie Mellon University, 1999.
- [43] Wiggins G. Papadopoulos G. A genetic algorithm for generation of jazz melodie solos. *Proceedings of STEP'98*, pages 7–9, 1998.
- [44] Wiggins G. Phon-Amnuasiuk S. The four-part harmonisation problem: A comparison between genetic algorithms and a rule-based system. In *Proceedings of AISB 99*, 1999.
- [45] Brown A. R. Opportunities for evolutionary music composition. *Proceedings of the ACMC'02*, pages 27–34, 2001.
- [46] Parncutt R. A perceptual model of pulse salience and metrical accent in musical rhythms. *Music Perception*, 11:409–464, 1994.

- [47] Chiu S.C. Shan M.K. Algorithmic compositions based on discovered musical patterns. *Multimedia Tools and Applications*, 46.
- [48] Unemi T. Sbeat3: A tool for multi-part music composition by simulated breeding. In *Proceedings of the Eighth International Conference on Artificial Life (A-Life VIII)*, 2002.
- [49] TEWI. Wirtualne laboratorium muzykologa, 2013. [dostęp online 07.03.2017: <http://tewi.p.lodz.pl>].
- [50] Setomoto M. Tominaga K. An artificial-chemistry approach to generating polyphonic musical phrases. In *Applications of Evolutionary Computing*, volume 4976 of *Lecture Notes in Computer Science*, pages 463–472. Springer Berlin Heidelberg, 2008.
- [51] Columbia University. Computer music center, 2017. [dostęp online 07.03.2017: http://cmc.music.columbia.edu/cmc_courses/].
- [52] Griffith University. Generative music (3710qcm), 2016. [dostęp online 07.03.2017: <https://degrees.griffith.edu.au/Course/3710QCM>].
- [53] Fayer M. Verbeurgt K., Dinolfo M. Extracting patterns in music for composition via markov chains. In *Proceedings of the 17th international conference on Innovations in applied artificial intelligence*, IEA/AIE'2004, pages 1123–1132. Springer Springer Verlag Inc, 2004.
- [54] Zabierowski W. Accord sequences generation in agreement with harmony tonal rules using cascade artificial neural networks. *Pomiary Automatyka Kontrola*, (11):1394–1396, 2011.
- [55] Napieralski A. Zabierowski W. Generation of harmonic sequences in accordance to tonal harmony rules with artificial neural networks. *Elektronika : konstrukcje, technologie, zastosowania*, 50(11):22–24, 2009.

17.05.2017

Masuranski Zdzisław



Gliwice, 24.04.2017

prof. dr hab. inż. Dariusz Kania
Politechnika Śląska
Instytut Elektroniki
ul. Akademicka 16
44-100 Gliwice

**Recenzja rozprawy doktorskiej mgr. inż. Łukasza Mazurowskiego dla
Rady Wydziału Informatyki Zachodniopomorskiego Uniwersytetu
Technologicznego**

(podstawą opracowania tej recenzji jest uchwała Rady Wydziału Informatyki Zachodniopomorskiego Uniwersytetu Technologicznego z dnia 23 marca 2017r.)

Tytuł rozprawy: Model systemu algorytmicznego komponowania muzyki na podstawie wzorców

Autor rozprawy: mgr inż. Łukasz Mazurowski

Promotor rozprawy: prof. dr hab. inż. Antoni Wiliński

Tematyka, cel pracy

Muzyka od wielu wieków jest nieodłącznym elementem naszej egzystencji. Czasami stanowi tylko tło codziennej aktywności, ale bywa również źródłem wielu inspiracji. Rozwój techniki komputerowej doprowadził do sytuacji, w której rolę wykonawcy, czy nawet kompozytora może pełnić komputer. Współpraca człowieka z komputerem odbywa się na różnych płaszczyznach. Komputer może stanowić wygodne narzędzie wspomagające proces twórczy (możliwość zapisywania, weryfikowania odsłuchowego fragmentów kompozycji), a nawet być elementem, który w procesie twórczym odgrywa rolę dominującą. Oczywiście, to człowiek tworzy oprogramowanie, które pozwala komputerowi tworzyć różnorodne utwory muzyczne. Nie jestem zwolennikiem zastępowania człowieka komputerem, przede wszystkim w obszarze związanym ze sztuką. W mojej opinii to nie jest droga, która może doprowadzić do stworzenia wybitnych dzieł, ale przytaczając myśl Chopina „Szczęśliwy ten, kto może być kompozytorem i wykonawcą zarazem” rozumiem zainteresowanie Autora, który realizując swoją pasję, mógł zetknąć się z dwojakim szczęściem, będąc równocześnie i kompozytorem, i szeroko rozumianym wykonawcą.

Istota pracy mgr. inż. Łukasza Mazurowskiego związana jest opracowaniem informatycznego modelu systemu pozwalającego na algorytmiczne komponowanie muzyki. Muzyka tworzona jest na podstawie wzorców, które odpowiednio rozbudowywane, modyfikowane, czy też sklejane tworzą większe formy muzyczne. Proces twórczy obejmuje dwa warianty komponowania utworów. W pierwszym kluczową rolę odgrywa sposób generowania wzorców muzycznych, które następnie są odpowiednio modyfikowane, przekształcane i sklejane, tworząc ostatecznie kompozycje o dość przypadkowej strukturze. W drugim wariacie, utwór muzyczny ma ściśle zdefiniowaną formę, która jest wypełniana odpowiednimi wzorcami muzycznymi. Podstawowym budulcem wzorca jest struktura

harmoniczna sekwencji akordów, do której dopasowywana jest linia melodyczna oraz uzupełniające całość partii instrumentalne i perkusyjne.

Tematykę rozprawy można umiejscowić w obszarze informatyki muzycznej, w której metody sztucznej inteligencji zostały wykorzystane do tworzenia oryginalnych utworów muzycznych. Proponowane rozwiązania ukierunkowane są na tworzenie muzyki generatywnej, która stanowi tzw. muzykę tła, wykonywaną przez różnego typu elektroniczne syntezatory dźwięku.

Oceniając obszar zainteresowań Pana mgr. inż. Łukasza Mazurowskiego stwierdzam, że wpisuje się on w ciekawy, choć mało popularny w Polsce nurt badań związanych z komputerowymi metodami komponowania utworów muzycznych. Warto jednak podkreślić, że jest on obiektem zainteresowania szeregu renomowanych uczelni technicznych, na czele z Massachusetts Institute of Technology. **Tematykę recenzowanej rozprawy doktorskiej należy uznać za interesującą, złożoną technicznie i aktualną.**

Zawartość, ocena merytoryczna rozprawy

Rozprawa Pana mgr. inż. Łukasz Mazurowskiego składa się z 5 zasadniczych rozdziałów oraz dodatkowych informacji zawartych w początkowym fragmencie pracy zatytułowanym „Charakterystyka pracy” oraz w dodatkach.

W rozdziale 1, po krótkim wprowadzeniu (podrozdział 1.1), w którym przedstawiono podstawowe encyklopedyczne informacje na temat dźwięku, struktury utworów muzycznych oraz metod ich reprezentacji, Autor koncentruje uwagę na różnorodnych, algorytmicznych koncepcjach tworzenia muzyki (podrozdział 1.2). Czyni to w sposób czytelny, choć niepotrzebnie ciągle odsyła czytelnika do dodatku zawierającego wydruki kodów źródłowych. W dalszej części (podrozdział 1.3) przedstawia i klasyfikuje komputerowe narzędzia do algorytmicznego komponowania muzyki. Całość rozważań, poparta szeregiem odwołań do literatury, świadczy o gruntownej i precyzyjnej analizie materiałów źródłowych. Rozdział 1 kończy fragment, w którym Autor formułuje cel pracy oraz hipotezę badawczą mówiącą, że *„bazując na metodach sztucznej inteligencji, możliwe jest opracowanie takich algorytmów komponowania utworów muzycznych, których efektem będą utwory atrakcyjne w odbiorze i posiadające strukturę poprawną w aspekcie teorii muzyki.”* Dziwi trochę bardzo nieśmiałe przedstawienie hipotezy badawczej, która wkomponowana w tekst nie została w żaden sposób wyróżniona. Rozumiem dylematy Autora, które niewątpliwie dotyczyły kwestii sformułowania tezy pracy. Zastosowane rozwiązanie nie uważam za niewłaściwe. Autor jasno mówi co chce osiągnąć, równocześnie uwypuklając problem obiektywnej oceny uzyskanego utworu muzycznego. Trzeba się z tym zgodzić, ponieważ odbiór muzyki ma niewątpliwie trudny do zmierzenia, subiektywny charakter. Znacznie większe wątpliwości budzi określenie zawarte w hipotezie badawczej, że efektem pracy są utwory *„posiadające strukturę poprawną w aspekcie teorii muzyki”*. W dalszej części wywodu mowa jest o zgodności utworów *„z kanonami teorii muzyki”* (str.48, l₂), lecz nie wiadomo o jakie dokładnie kanony chodzi. W tej sytuacji trudno ocenić, czy uzyskane wyniki są zgodne z owymi kanonami. Problem w tym, że skomponowane utwory bardzo często znacząco odbiegają od elementarnych zasad klasycznej harmonii, które powinny stanowić nienaruszalny fundament tworzący *„kanony teorii muzyki”* (wyjaśnienie w części uwagi krytyczne, wątpliwości, pytania, p. 1).

W rozdziale 2 przedstawiono koncepcje algorytmicznych systemów komponowania muzyki. Po krótkim wprowadzeniu (podrozdział 2.1), w którym zawarto ogólne spojrzenie na proces komponowania, zaprezentowano dwa autorskie warianty procesu generacji utworów muzycznych. W pierwszej metodzie, proces komponowania opiera się na tworzeniu krótkich wzorców muzycznych, które w dalszej kolejności łączone są w większą całość. Autor precyzyjnie przedstawia kolejne etapy tworzenia utworu, koncentrując uwagę na

zastosowanych rozwiązaniach, a całość rozważań ubiera w na ogół czytelne ramy formalne. Nie da się ukryć, że koncepcję generacji muzyki na podstawie wzorców cechuje szereg ograniczeń. Wydaje się, że największym ograniczeniem jest oderwanie procesu tworzenia utworu od powszechnie występujących kanonów harmoniczych. Zastosowany w algorytmach, sposób określania funkcji harmoniczej na podstawie profili Krumhansla-Kesslera, w mojej opinii nie końca precyzyjnie przedstawiony, (*Uwagi krytyczne, pytania, wątpliwości*, p. 2) z całą pewnością jest dość daleko oderwany od „niekomputerowego” komponowania muzyki. Tym bardziej doceniam próbę eliminacji tego braku, która wraz z możliwością określania struktury tworzonego działu muzycznego o wyraźnie wyodrębnionych częściach, stanowi istotę rozważań zaprezentowanych w podrozdziale 2.4. Odmienność proponowanych metod sprowadza się między innymi do sposobu generowania wzorców muzycznych. Tym razem bazę twórczą stanowią sekwencje akordów, do których dopasowywana jest linia melodyczna oraz linia basowa. Uzupełnieniem treści zawartych w rozdziale 2 jest krótki podrozdział 2.4 na temat instrumentacji i syntezy kompozycji, który nie pozostawia niedomówień i jasno precyzuje, co stanowi podstawę dociekań naukowych stanowiących istotę pracy.

Struktura ramowa rozdziału 3 jest identyczna jak rozdziału 2. Kolejno omawiane są implementacje poszczególnych koncepcji w środowisku Matlab z wykorzystaniem elementów dedykowanych do obsługi MIDI (MIDI Toolbox). W pierwszej części (podrozdział 3.1) uwagę skoncentrowano na metodzie tworzenia utworów bez definiowania struktury powstającej kompozycji, a w drugiej (podrozdział 3.2) na metodzie komponowania, w której nadrzędną zasadą jest wstępne określenie struktury utworu. W końcowym, krótkim podrozdziale 3.3 wskazano na możliwość końcowego modyfikowania uzyskanych wyników kompozycji, polegającą na wyborze odpowiednich instrumentów, czy też parametrów użytych syntezy dźwięku. Istota tego etapu kompozycji sprowadza się do „ręcznej” modyfikacji plików MIDI, czy też doboru gotowych brzmień zawartych w bankach brzmień wykorzystywanych syntezy.

W rozdziale 4 przedstawiono wyniki przeprowadzonych eksperymentów. Tak jak poprzednio rozdział podzielono na dwie części, związane z proponowanymi metodami komponowania. W kolejnych podrozdziałach omówiono wpływ zmian poszczególnych parametrów na uzyskiwany efekt finalny. Strukturę uzyskanych utworów przedstawiono w postaci wykresów pianolowych. Uzupełnieniem zawartych w poszczególnych podrozdziałach rozważań są pliki muzyczne z zapisem kompozycji w formacie flac zawarte na płycie CD. Mimo, iż treść rozdziału jest przesyciona drobiazgowym omawianiem często subtelnie zmieniających się parametrów, całość rozważań przeczytałem z zainteresowaniem, tym bardziej, że wszystkie wnioski mogłem „usłyszeć”, słuchając zamieszczonych na płycie kompozycji. Mimo wszystko czuję pewien niedosyt, ze względu na zbyt techniczny charakter rozważań. Mam świadomość, że aspekty informatyczno-techniczne są najważniejsze w tego typu opracowaniu, ale w mojej opinii zupełnie zagłuszają czysto muzyczne podłoże zaproponowanych rozwiązań. Każda z wyodrębnionych części rozdziału 4 kończy się krótkim podsumowaniem (podrozdziały 4.1.5 oraz 4.2.7), w których w syntetyczny sposób podsumowano uzyskane wyniki.

Uzupełnieniem treści pracy jest rozdział 5, w którym nakreślono różnorakie obszary działalności naukowej i popularyzatorskiej Autora, ściśle związane z tematyką recenzowanej pracy oraz przedstawiono wizję platformy systemu algorytmicznego komponowania muzyki wykorzystującą osiągnięcia naukowe Autora. W kończącym rozdział 5 podrozdziale pt.: „Wnioski końcowe” zawarto uzasadnienie potwierdzające, że cele pracy zostały osiągnięte, a hipoteza badawcza została udowodniona. Czytając zakończenie pracy ponownie odczuwam pewien dyskomfort związany z brakiem precyzji w przedstawieniu hipotezy badawczej. O ile mogę przymknąć oko na ocenę atrakcyjności uzyskanej muzyki, to brakuje

mi precyzyjnego określenia jej „poprawności w aspekcie teorii muzyki”. W zakończeniu Autor stwierdza (str. 144, 1⁷), że „W procesie komponowania starannie przestrzegano zasad poprawności tworzenia muzyki według jej kanonów historycznie utrwalonych...”, natomiast jak już wspomniałem wcześniej, nigdzie nie precyzuje o jakie dokładnie kanony Mu chodzi. Powyższy brak uważam za dość istotny, ponieważ szereg podstawowych, historycznie utrwalonych zasad harmonii z całą pewnością nie zostało dotrzymanych, co znacząco zmniejsza wartość muzyczną uzyskanych kompozycji (*uwagi krytyczne p. 1*).

Dopełnieniem treści pracy jest wykaz literatury obejmujący 118 pozycji, glosariusz, spis rysunków, tablic oraz dodatki zawierające kody źródłowe oraz informacje na temat zawartości dołączonej płyty CD. Szkoda, że oprócz plików w formacie flac nie zamieszczono opisu kompozycji w postaci plików MIDI. Tego typu dane stanowiłyby uzupełnienie informacji zawartych na często nieczytelnych wykresach pianolowych.

Zaprezentowane w pracy koncepcje algorytmicznych metod komponowania muzyki uważam za ciekawe. Odnosząc zaproponowane rozwiązania do sposobów aranżacji, czy też instrumentalizacji utworów, można by stwierdzić, że proponowane metody różnią się kierunkiem powstawania dzieła. W pierwszym przypadku, który można kojarzyć z komponowaniem poziomym, ważniejszy jest „poziomy” (w sensie partytury) aspekt tworzenia linii melodycznych, niż sposób uzyskiwania towarzyszących melodii współbrzmień. W drugim podejściu, zapominając o narzuconej na wstępie formie całego utworu, mamy do czynienia z pionowym kierunkiem powstawania dzieła. Tym razem linie melodyczne są dopasowywane do podstawy harmonicznego utworu.

Pewne wątpliwości mogą budzić znaczące ograniczenia komponowania muzyki metodą poziomą, zwaną w pracy metodą bez określenia formy utworu, ponieważ prowadzą one do bardzo przypadkowej postaci utworu. W drugiej koncepcji komputerowego komponowania, kojarzonej przeze mnie z metodą komponowania pionowego, z całą pewnością drzeźnią większe możliwości. Dzieje się tak ze względu na bardziej naturalny sposób tworzenia wzorców oraz mniej przypadkowe ich łączenie w większą całość.

Nie sposób nie docenić dużego wkładu pracy Doktoranta w opracowanie algorytmicznych metod kompozycji, zmaterializowanych w postaci algorytmów opisanych w środowisku Matlab oraz wykonanie licznych eksperymentów, które stanowią ostateczny wynik opracowanych metod. Uzyskane efekty są interesujące, choć po drobnych modyfikacjach procesu twórczego mogłyby poszerzyć grono zwolenników muzyki tworzonej komputerowo (*szczegóły uwagi krytyczne p. 1*).

Do najważniejszych osiągnięć zaliczam:

1. Opracowanie kompleksowych koncepcji komponowania muzyki na podstawie wzorców w dwóch wariantach: bez określenia formy utworu oraz ze zdefiniowaną strukturą utworu.
2. Opracowanie algorytmicznych metod generowania wzorców muzycznych, ze szczególnym uwzględnieniem generowania wzorców przy zadanej strukturze harmonicznego.
3. Opracowanie i zaimplementowanie w środowisku Matlab algorytmicznych metod komponowania muzyki na podstawie wzorców,
4. Przeprowadzenie licznych eksperymentów potwierdzających możliwość przeprowadzenia całego procesu algorytmicznego komponowania utworów.

Uwzględniając powyższe, temat recenzowanej rozprawy doktorskiej mgr. inż. Łukasza Mazurowskiego uważam za interesujący i uzasadniony technicznie. Stopień złożoności, znaczenie naukowe i zakres zadania odpowiadają ustawowym i zwyczajowym wymogom stawianym rozprawie doktorskiej. Poziom merytoryczny w obszarze informatyczno-technicznym nie budzi zastrzeżeń.

Podsumowując ten fragment recenzji stwierdzam, że cel pracy został osiągnięty. **Wyniki przedstawionych w pracy eksperymentów potwierdzają możliwość opracowania algorytmów komponowania utworów muzycznych, których efektem są interesujące, choć dość prymitywne w swojej strukturze harmoniczej utwory muzyczne. W świetle faktu, że każda ocena dzieła muzycznego jest subiektywna, można stwierdzić, że uzyskany efekt nie podważa, a raczej potwierdza prawdziwość hipotezy badawczej.**

Rozprawa doktorska ma również słabe strony i różnego typu niedociągnięcia. Podzieliłem je na dwie grupy: uwagi krytyczne, wątpliwości, pytania, w większości dotyczące aspektów muzycznych oraz uwagi szczegółowe, zawierające niedociągnięcia językowe i edycyjne.

Ogólnie rzecz ujmując, mam szereg uwag, wątpliwości, dotyczących sposobu przedstawienia aspektów muzycznych, które bezpośrednio wpływają na niedoskonałość uzyskanych wyników komponowania. Mam świadomość, że muzyczne niedociągnięcia daleko wybiegają poza kwestie natury technicznej, ale czy w tego typu pracy są one nieistotne? Analizując pracę niezwykle przeszkadzała mi przytłaczająca ilość nie zawsze potrzebnych formalizmów, które zupełnie zagłuszyły muzyczne aspekty przyjętych rozwiązań. Większość rozważań przedstawiona jest z pominięciem odniesień do muzycznych fundamentów zaprezentowanych koncepcji. Mam świadomość, że recenzowana praca jest dziełem z obszaru informatyki, a nie muzyki, ale mimo wszystko uważam, że uwypuklenie muzycznych inspiracji w procesie przedstawienia algorytmów nie byłoby wadą opracowania, a pozwoliłoby lepiej zrozumieć istotę zaproponowanych pomysłów.

Chciałbym podkreślić, że znacznie wyżej cenię osiągnięcia natury informatycznej, niż uzyskane utwory muzyczne. Część moich uwag ma charakter dyskusyjny, a ich podłoże może wynikać z subiektywnej oceny skomponowanych utworów. Mam nadzieję, że wskazane niedociągnięcia staną się dla Doktoranta inspiracją, która pozwoli w przyszłości udoskonalić algorytmy komponowania muzyki.

Uwagi krytyczne, wątpliwości, pytania

1. W wielu miejscach pracy (również w hipotezie badawczej) podkreślano, że komponowane utwory powinny mieć „*strukturę poprawną w aspekcie teorii muzyki*” (hipoteza badawcza). Nigdzie precyzyjnie nie zdefiniowano „*kanonu teorii muzyki*” (str.48, l₂), co budzi liczne wątpliwości, tym bardziej, że delikatnie rzecz ujmując, struktura uzyskanych utworów bardzo często drażni wieloma muzycznymi niedociągnięciami. Chciałbym zwrócić uwagę na dwa fakty, które stanowią niedociągnięcia, zdecydowanie obniżające wartość skomponowanych utworów. Rozważmy przykładowo wzorzec f₄₄₄₄, którego struktura akordów opisana jest w trzecim wierszu tabeli 4.12. Akordy występują zawsze w pierwszym przewrocie G₂#,B₂,D₃#: C₃#,E₃,G₃#: D₃#,F₃#,A₃#. Konsekwencją użycia tylko pierwszego przewrotu akordów oraz skupionego układu toniki (T), subdominanty (S) i dominanty (D) jest wszechobecne występowanie w ciągu używanych współbrzmień kwint równoległych, zakazanych w harmonii klasycznej. Obecność kwint równoległych w postęпах akordowych z całą pewnością zdecydowanie obniża atrakcyjność uzyskiwanych utworów. Podobnie dużym problemem jest niewłaściwe rozwiązywanie akordu dominanty

systemowej $D_3\#, F_3\#, A_3\#, C_4\#$ na przykład w utworze f_1234. Dominanta septymowa jest akordem budującym napięcie i takim, który wymusza odpowiednie rozwiązanie. Nie bez przyczyny jej obecność zwiastuje zwykle zakończenie całego utworu, czy też koniec krótszej myśli muzycznej. W rozważanej kompozycji następstwem dominanty septymowej ($D_3\#, F_3\#, A_3\#, C_4\#$) jest tonika ($G_2\#, B_2, D_3\#$), ale zastosowany przewrót toniki powoduje, że odpowiednie dźwięki składowe akordu tonicznego nie pojawiają się tam gdzie trzeba. Tego typu sytuacja jest bardzo daleka od klasycznego kanonu powszechnie stosowanego w większości utworów muzyki klasycznej, rozrywkowej, filmowej itp. Chciałbym podkreślić, że powyższe braki można stosunkowo łatwo wyeliminować wprowadzając do założonej struktury akordów ich przewroty i zapominając o kurczowym trzymaniu się układów skupionych. Wystarczyłoby wskazać sekwencję akordów $T \rightarrow S \rightarrow D$ ($G_2\#, B_2, D_3\# \rightarrow C_3\#, E_3, G_3\# \rightarrow D_3\#, F_3\#, A_3\#$) lekko zmodyfikować i przedstawić na przykład w postaci $G_2\#, B_2, D_3\# \rightarrow G_2\#C_3\#, E_3 \rightarrow F_2\#, A_2\#, D_3\#$, w której drażniące słuchacza kwinty równoległe zostały wyeliminowane. Tego typu zmiany nie powinny wprowadzać większych komplikacji w procesie tworzenia wzorców muzycznych generowanych na podstawie sekwencji akordów. Gdybym się jednak mylił, to zawsze można końcowy efekt kompozycji poddać korekcie, wyłapując niewłaściwe postępy interwałowe i obarczone wadami rozwiązania akordów dominanty septymowej.

2. W algorytmie komponowania bez określenia struktury utworu jednym w kluczowych problemów jest algorytm generowania wzorców muzycznych. W podrozdziale 2.2.1 przedstawiono opis reguł generowania wzorców. W regułach gr3, gr4 pojawia się problem określenia tonacji utworzonego fragmentu, do rozwiązania którego wykorzystano znaną z literatury metodę Krumhansla-Kesslera. Nasuwa się jednak szereg wątpliwości. Metoda Krumhansla-Kesslera pozwala na określenie tonalności utworu poprzez analizę występujących w nim dźwięków, sprowadzającą się do określania wartości współczynnika korelacji liniowej Pearsona pomiędzy dźwiękami zawartymi w utworze a tzw. profilami tonacji. Wyniki eksperymentów przedstawione w pracy [31] pokazują, że tonację można określić na podstawie zaledwie kilku dźwięków. Oczywiście skuteczność określenia tonalności utworu jest tym większa im dłuższy fragment utworu podlega analizie. W tej sytuacji pojawia się szereg pytań, na które nie znalazłem precyzyjnej odpowiedzi. Analiza odsłuchowa załączonych na płycie CD kompozycji również nie rozwiała moich wątpliwości. Na podstawie jakiej próbki określona jest tonalność wzorca? Czy dany wzorec kojarzony jest zawsze tylko z jednym akordem, będącym dla niego akordem tonicznym? Czy tworząc utwory próbowano określać zasady konkatencji wzorców muzycznych na podstawie ich tonalności?
3. W algorytmie komponowania z określoną strukturą utworu ważnym aspektem jest sposób generacji wzorców, który polega na dopasowaniu linii melodycznych do założonej sekwencji akordów. Tego typu zasada może prowadzić do znacznie atrakcyjniejszych utworów od utworów komponowanych pierwszą metodą, cechujących się wszechobecną przypadkowością. Niestety przedstawiony w pracy opis nie wyjaśnia wszystkich aspektów proponowanego sposobu postępowania. Przykładowo, nie wiem czemu miało służyć przedstawienie w pracy tabeli 2.7, w której w poszczególnych kratkach zawarto symbole $p_{11}, p_{12}, \dots, p_{77}$. W końcowym fragmencie tekstu na str. 68 zawarto informację, że wartości p_{ij} określają prawdopodobieństwa wystąpienia akordu i -tego stopnia bezpośrednio przed akordem zbudowanym na j -tym stopniu gamy. Nie wiadomo jakie są te prawdopodobieństwa, a przecież od nich w istotny sposób zależy struktura harmoniczna utworu, a co za tym idzie jego atrakcyjność. Dodatkowym mankamentem jest niejasność w oznaczeniach stopni, na których budowane są akordy. W tabeli 2.6 pojawiają się oznaczenia ii, vi, a tabeli 2.7 z takich oznaczeń zrezygnowano. Z danych zawartych w tab.

- 2.6 wynika, że chodzi o akordy paralelne do toniki (a-moll) oraz subdominanty (d-moll). Co w takim razie oznaczają symbole II, VI, III zawarte w tab. 2.7? Analiza odsłuchowa dołączonych na płycie utworów jednoznacznie wskazuje na obecność akordów septymowych, o których nie wspomniano w opisie algorytmów. Pojawiły się dopiero w tabeli 4.12 (dis⁷). W jaki sposób one powstają? Czy w tej sytuacji tabela 2.6 zawiera wszystkie wykorzystywane sekwencje akordów? A może powstający akord dominanty septymowej jest wynikiem losowego połączenia dźwięku linii instrumentalnej z bazowym akordem dominanty występującym w użytej sekwencji akordów, ale w tej sytuacji nie powinien on się znaleźć w tabeli 4.12?
4. Dużym utrudnieniem analizy pracy jest niekonsekwentny sposób opisu dźwięków H i B, którego geneza tkwi w różnicy oznaczeń używanych w literaturze polskiej, niemieckiej i anglojęzycznej. W początkowej części pracy symbol H oznacza dźwięk leżący sekundę małą niżej niż C (str. 60, 61, 62). W rozdziale 4 (str. 114) mowa jest już o akordzie molowym G₃#, B₃, D₄#, który może być akordem tonicznym gis-moll, tylko w sytuacji, gdy symbol B₃ oznacza dźwięk leżący sekundę małą niżej niż C₄. Informacja na temat przyjętej konwencji opisu dźwięków, niejednolitej w całej pracy, pojawia się dopiero na str. 117, l₁₆.
 5. W pracy występują błędy, niejednoznaczności wynikające z niekonsekwentnego stosowania zasad enharmonii. Przykładowo na str. 121 (l₅) pojawia się opis akordu durowego w pierwszym przewrocie (G₃#, C₄, D₄#), który przyjmując zastosowany opis składa z dwóch interwałów: kwarty zmniejszonej (G₃#, C₄) oraz sekundy zwiększonej (C₄, D₄#). Jeżeli na siłę założymy, że chcemy określić dźwięki występujące w akordzie tonicznym Gis-dur, to przyjmując konwencję amerykańską powinniśmy opisać jego postać jako (G₃#, B₄, D₄#). Naturalniej byłoby opisać to samo współbrzmienie jako (As₃, C₄, Es₄). Opis (As₃, C₄, Es₄) stanowi toniczny akord As-dur w pierwszym przewrocie, identycznie brzmiący jak akord Gis-dur, co można potwierdzić na przykład patrząc na strukturę koła kwintowego.
 6. Analizę treści pracy utrudnia brak dbałości o wyczerpujący opis rysunków i danych zawartych w tabelach. Przykładowo tabela 2.3 zawiera dwa zestawy suchych liczb. Zdaję sobie sprawę, że szczegółowy opis zawarto w tekście, ale uważam, że wprowadzenie opisu kolumn, zawierającego nazwy dźwięków oraz skojarzenie wierszy z nazwami tonacji (durowych i molowych) zdecydowanie poprawiłoby czytelność danych zawartych w tabeli. Dodatkowo, w tekście bezpośrednio pod tabelą 2.3 niewłaściwie określono uporządkowanie danych mówiąc, o kolejności „od prawej strony”. Prawdopodobnie podano mylny kierunek, ponieważ dla tonacji C-dur (pierwszy wiersz) wartość współczynnika skojarzonego z dźwiękiem C powinna być znacznie większa od wartości skojarzonej z dźwiękiem H (konwencja polska), Fis itd.

Uwagi szczegółowe

1. W pracy pojawiają się bardzo niefortunne, czasami żargonowe sformułowania:
 - str. 6, l₅ (l₅ - linia 5 od dołu) - „Jako wysokość dźwięku należy rozumieć częstotliwość z jaką występuje.”;
 - str. 7, l¹⁻², „...cała nuta dzieli się na dwie **pół nuty**, następnie **półnuta** dzieli się...”; podobnie str. 53, l₈,
 - str. 60, l¹³ - „...z uwzględnieniem długości nuty”; Nuta określa wysokość, a dopiero po uwzględnieniu tempa, długość trwania dźwięku. W tej sytuacji trudno mówić o długości nuty. Należało użyć określenia długość dźwięku.

- str. 60, l²⁻³ - „... y stanowi listę ważonych wartości w dwunastostopniowej skali w danej tonacji.” Z powyższego sformułowania można wnioskować, że wartości y określone są w dwunastostopniowej skali. Dwunastostopniowa skala jest materiałem dźwiękowym, z którego odpowiednio wybranych 7 dźwięków stanowi podstawowy materiał dźwiękowy utworu utworzonego w odpowiedniej tonacji.
- str. 126, l₂₅₋₂₆ - „motyw muzyczny *wchodzący w dysonans* z generowaną melodią”, żargon, powinno być „tworzący dysonanse”

2. Niewłaściwe użycie, rozumienie pojęć muzycznych

- str. 11, l⁵⁻⁷ - „Nakładanie się dźwięków w czasie jest domeną polifonii i to właśnie ona występuje w niemalże każdym współczesnym utworze muzyki rozrywkowej.” Trudno zgodzić się ze stwierdzeniem, że muzyka rozrywkowa jest oparta na polifonii. Jest wręcz odwrotnie. Dominującą formą muzyki rozrywkowej jest faktura homofoniczna, w której można wyróżnić wyrazistą melodię i towarzyszący jej akompaniament. Muzyczne znaczenie słowa „polifonia” określa fakturę muzyczną utworu, który składa się zwykle z dwóch lub trzech niezależnych linii melodycznych. W znaczeniu technicznym pojęcie „polifonii” kojarzone jest z możliwością równoczesnego wydobywania kilku dźwięków. Autor w pracy nie rozróżnia tych dwóch istotnie różnych znaczeń, co w przypadku pracy dotyczącej komponowania muzyki z całą pewnością jest niewłaściwe. Podobne niewłaściwe użycie pojęcia polifonii występuje na str. 22, l₇ oraz str.57, l₃.
- str. 108, l₂₋₃ - „...zmiana...długości...dźwięku powoduje zmianę w dynamice całej kompozycji...” Słowo „dynamika” w języku muzycznym ma inne znaczenie. Cały utwór może być skomponowany w jakimś odcieniu dynamicznym (np. piano, forte, fortissimo itd.) lub jego dynamika może się zmieniać w czasie, raptownie bądź stopniowo. Potoczne sformułowanie „utwór dynamiczny” należało zastąpić określeniem „utwór w szybkim tempie”.

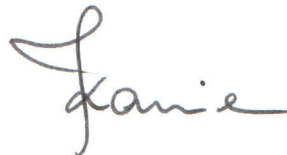
3. Literówki, drobne niedociągnięcia natury edycyjnej:

- str. 10, l³, str. 40, l¹⁰, str.53, l⁶, str.53, l₁₉ – powinno być „zarówno”,
- str. 13, l³ - brak wyjaśnienia skrótu „bpm”; wyjaśnienie umieszczono dopiero na str. 18,
- str. 24, tab. 1.7- nałożenie zawartości dwóch kolumn,
- str. 28, l¹⁶, str. 110, l₁₆, - zamiast „wypadku”, powinno być „przypadku”,
- str. 37, l₁₆ – „Przez ...?... oznacza się...”,
- str. 37, l₅ – „...siatki 1 i 2 obu automatów...”,
- str. 39, l₇, l₃, l₂ - błędne odwołanie do rysunku,
- str. 67, l₁ – zamiast „następczości” powinno być „następstwo”,
- str. 82, l₂ – „na stronach 60 – 60”,
- str. 98, l³ – zamiast „macierz” powinno być „macierzy”,
- str. 103, l₄ – W tab. ?? znajdują się....
- str. 114, l₈ – zamiast „k_raw_421.flac” powinno być „k_raw_411.flac”,
- str. 126, l₂₄ – zamiast „ten koncept” lepiej byłoby „ta koncepcja”
- str.130, l¹⁴ – zamiast „znanie” powinno być „znaczenie”,
- str.132, l₄ – zamiast „cechą” powinno być „cechom”,
- rys.5.3-5.7 są zupełnie nieczytelne,
- str. 3, l₁₃, str. 9, l₃, l₁₉, str. 10, l⁴, itd. – przykładowe niedociągnięcia interpunkcyjne.

Należy jednoznacznie stwierdzić, że powyższe uwagi szczegółowe wyspecyfikowane z opiniotawczego obowiązku, nie obniżają wartości merytorycznej rozprawy.

Ocena końcowa rozprawy

Podsumowując stwierdzam, że mimo pewnych wad, w większości dotyczących aspektów muzycznych, rozprawa doktorska Pana mgr. inż. Łukasza Mazurowskiego spełnia wymagania stawiane w "Ustawie o stopniach naukowych i tytule naukowym oraz o stopniach i tytule w zakresie sztuki", zatem **wnoszę o dopuszczenie rozprawy doktorskiej do publicznej obrony.**

A handwritten signature in black ink, appearing to read 'Janie'.

Szczecin, dnia 17 maja 2017 r.

dr hab. inż. Przemysław Klęsk, prof. nadzw.
Katedra Metod Sztucznej Inteligencji i Matematyki Stosowanej
Wydział Informatyki
Zachodniopomorski Uniwersytet Technologiczny w Szczecinie
ul. Żołnierska 49
71-210 Szczecin

RECENZJA

rozprawy doktorskiej mgr inż. Łukasza Mazurowskiego
pt.: „Model systemu algorytmicznego komponowania muzyki
na podstawie wzorców”

1 Zawartość rozprawy i problem badawczy

Recenzowana rozprawa wpisuje się tematycznie w zakres tzw. sztucznej twórczości, a dokładniej sztucznego komponowania muzyki. Inne przykłady sztucznej twórczości to m.in. sztuczna poezja (czyli generowanie z pomocą komputera form poetyckich) lub też sztuczne malarstwo (komputerowe generowanie obrazów przypominających formy malarskie). Sztuczna twórczość to zdecydowanie dziedzina niszowa, którą można zaliczyć do pobocznych nurtów w ramach sztucznej inteligencji.

Należy z góry zaznaczyć, że w tego typu zagadnieniach nie jest łatwo o jednoznaczne ilościowe sposoby oceny otrzymanych rozwiązań. Zwyczajowo przyjmuje się *niestety* pewien subiektywny sposób oceny. Z drugiej strony warto sobie uzmysłwić, że na wygenerowane sztuczne utwory można patrzeć poprzez pryzmat szczególnego rodzaju *testu Turinga* (lub też gry w naśladownictwo). Można zadać sobie np. pytanie: co, jeśli pewien sztucznie skomponowany utwór muzyczny byłby w stanie „oszukać” (co do natury swojego powstania) odpowiednio duży procent muzyków? Tak postawione zagadnienie jest już moim zdaniem ciekawe badawczo.

Autor w pracy zajmuje się tzw. *muzyką tła* (ang. *background music*). Chodzi tu o muzykę, która może być odtwarzana podczas wykonywania pewnych czynności, a która nie angażuje mocno uwagi słuchaczy np. muzyka odtwarzana w sklepach, korytarzach hotelowych, itp. Przyuszczalnie zabieg ograniczenia się w pracy do muzyki tła ma na celu odpowiednie uproszczenie problemu. Nie należy za to jednak zbytnio krytykować autora. Jest intuicyjnie jasnym, że „ucho ludzkie” ma dużo wyższe wymagania (melodyczne, rytmiczne, itd.) np. odnośnie piosenki, aniżeli odnośnie improwizacji jazzowej.

Za cel pracy autor postawił: „(...) opracowanie modelu komputerowego systemu komponowania muzyki użytkowej (posiadającej wartość rynkową), w którym do generowania oryginalnych utworów muzycznych wykorzystuje się metody autorskie oparte na metodach sztucznej inteligencji (...)”.

Postawiona w pracy hipoteza badawcza brzmi następująco: „(...) bazując na metodach sztucznej inteligencji, możliwe jest opracowanie takich algorytmów komponowania utworów muzycznych, których efektem będą utwory atrakcyjne w odbiorze i posiadające strukturę poprawną w aspekcie teorii muzyki.”. O ile pierwsza własność — atrakcyjność w odbiorze — jest, tak jak

wcześniej zaznaczono sprawą subiektywną (lub wymaga poddania pod głosowanie), tak druga własność — poprawna struktura muzyczna — jest moim zdaniem słusznie wskazana i uporządkowuje trochę tok badań.

2 Ogólna ocena struktury pracy

Praca liczy 194 ponumerowane strony. Jest zbudowana z: dwustronicowej „Charakterystyki pracy” oraz 5 rozdziałów właściwych (w tym piąty konkludujący).

Rozdział pierwszy „Komponowanie algorytmiczne muzyki — wprowadzenie i analiza rozwiązań” wprowadza czytelnika w zagadnienia muzyczne, dokonuje przeglądu metod i systemów komputerowych do generowania muzyki i formułuje problem badawczy.

Rozdział drugi „Koncepcja systemu algorytmicznego komponowania muzyki” przedstawia autorskie pomysły mgra Mazurowskiego dotyczące algorytmicznego komponowania z dwiema wyraźnie wydzielonymi częściami: (1) komponowanie bez definiowania formy generowanego utworu, (2) komponowanie z uwzględnieniem opisu formy generowanego utworu.

Rozdział trzeci opisuje szczegóły implementacji (skrypty w środowisku MATLAB). Implementacja podobnie jak koncepcja z rozdziału drugiego jest podzielona na dwie części — komponowanie bez formy i z uwzględnieniem formy.

Rozdział czwarty to rozdział eksperymentalny.

Rozdział piąty przedstawia końcowe konkluzje, perspektywy rozwoju systemu i komentarze na temat dorobku publikacyjnego i pozapublikacyjnego autora.

Praca została złożona w systemie L^AT_EX. Zawiera bibliografię (108 pozycji), odpowiednie spisy rysunków i tablic, pomocny glosariusz pojęć muzycznych, dodatek z listingami kodów źródłowych oraz dodatek z opisem zawartości załączonej płyty CD. Strukturę pracy należy ogólnie uznać za przemyślaną i staranną, natomiast odbiór pracy przez czytelnika niestety obniża duża liczba błędów edytorskich, matematycznych, notacyjnych i innych (wymienionych szczegółowiej w punkcie 6 tej recenzji).

3 Metody zaproponowane przez autora

Autor przedstawia dwa zasadnicze pomysły dotyczące komponowania algorytmicznego, kolejno: bez uwzględniania formy wynikowego utworu i z jej uwzględnieniem.

W pomysłu pierwszym główną informacją zasilającą proces komponowania (pochodzącą od człowieka, użytkownika programu) jest, mówiąc w uproszczeniu, para: motyw, rytmika. Motyw podstawowy jest reprezentowany przez wektor wysokości dźwięków (bez ich czasów trwania), natomiast rytmikę opisują wektory `rhythmic_step` i `rhythmic_value`, które przekładają się dalej na wektor możliwych pozycji nut, na których osadzany będzie motyw lub jego wariacje. Proces komponowania zaproponowany przez autora można wówczas streścić następująco. Na podstawie motywu podstawowego tworzona jest macierz wariacji¹ tego motywu (macierz KMM), powstała np. poprzez permutacje lub transformacje (np. przesunięcia o 4 półtony, 5 półtonów; przesunięcia rotacyjne, itp.). Dalej, powoływany jest do życia zestaw tzw. *mini-generatorów* służących do generowania tzw. *wzorców*. Mini-generatory to krótkie parametryzowane procedury narzędziowe, które można uznać za serce pomysłu autora i cegiełki budujące wynikową kompozycję. Parametry mini-generatora to m.in.: klasa (monofoniczny, polifoniczny, perkusyjny), używany motyw muzyczny (wiersz wybrany z macierzy KMM), wektor pozycji nut, minimalna długość nuty generowanego wzorca, długość wzorca w jednostkach bpm, tempo wzorca w jednostkach

¹Autor nazywa ją właściwie macierzą kombinacji motywów (KMM), przy czym w zależności od kontekstu autor mówi także o permutacjach motywu lub transformacjach motywu.

bpm. Wygenerowane wzorce są dalej sklejane z pewną losowością w wynikową kompozycję, przy czym możliwość sklejania dwóch wzorców jest uzależniona od odległości ich histogramów wysokości dźwięków (z uwzględnieniem modelu akustycznego Parncutta). Autor posługuje się odległością euklidesową. Warto dodać, że w przypadku mini-generatorów polifonicznych autor zauważa problem dbałości o zgodność tonacji i wykorzystuje metodę Krumhansla-Schmucklera opartą na współczynniku korelacji dla dwunastopniowych skali tonacji.

Drugi pomysł autora dotyczy komponowania z uwzględnieniem formy muzycznej. Chodzi tu o generowanie utworów, które miałyby pewną strukturę zamkniętą np. posiadającą wstęp, rozwinięcie i zakończenie, z uwzględnieniem możliwych powtarzalności pewnych fragmentów. Tutaj również można dopatrzeć się głównej informacji zasilającej (podawanej przez człowieka) w postaci pary: struktura formy, akord bazowy. Opisem struktury formy jest wektor (np. `form = [1, 1, 2, 3, 2, 3, 1]`), wskazujący unikalne fragmenty i kolejność ich występowania. Widać tu pewną analogię np. do schematów rymów w poezji (np. rymy AABB lub ABAB, itp.). Akord bazowy stanowi natomiast informację początkową do budowania tzw. progresji akordów i tym samym harmonii. Progresja jest wytyczna w sposób probabilistyczny na podstawie reguł kadencji muzycznych poprzez zbudowanie macierzy przejść akordów (silna analogia do macierzy prawdopodobieństw przejść pomiędzy stanami w ukrytych modelach Markowa). Mając sekwencję akordów, generowana jest następnie (z pewną losowością) melodia i linia basowa. Rola mini-generatorów w tym podejściu została ograniczona do generowania partii perkusyjnych.

4 Główne uwagi do rozprawy

4.1 Pozytywne strony rozprawy

1. Obie proponowane koncepcje algorytmiczne autora (tzn. podejścia do komponowania bez formy i z jej uwzględnieniem) recenzent uznaje za przemyślane i ciekawe.
2. Na pochwałę zasługują dokładne studia literaturowe przeprowadzone przez autora. W pracy dokonano przeglądu różnych technik i prób algorytmicznego komponowania muzyki sięgających nawet do XVII wieku. Opisane są pomysły teoretyków muzki i matematyków m.in.: Mersenne'a, Kirchera, Kirnbergera, Himmela. Więcej miejsca autor poświęca idei serializmu Schoenberga, metodzie formalizmu Xenakisa, a także komponowaniu z pomocą: modeli Markowa, gramatyk formalnych, automatów skończonych, algorytmów genetycznych, automatów komórkowych, czy wreszcie sieci neuronowych. Jest to szeroki i dojrzały przegląd, w którym widać dobre rozumienie autora omawianych treści.
3. Praca jest bogata pod względem eksperymentalnym. W rozdziale czwartym autor omawia wpływ zmian wielu przewidzianych przez niego parametrów i nastaw na uzyskiwane kompozycje. Chodzi tu m.in. o takie parametry jak: długość dźwięku, długość wzorca, parametry zbioru wygenerowanych wzorców, tablica powtórzeń, wektor opisu struktury (formy), wektor skali muzycznej, wektory lokalizacji i przesunień dźwięków, liczby akordów i ich czasy trwania.
4. Dzięki płycie CD dołączonej do pracy można przesłuchać otrzymane przez programy autora kompozycje, a także ich części składowe (wzorce, akordy, itp.). Jest to ciekawy materiał ponad 60 plików z nazewnictwem powiązanym z odpowiednimi parametrami, które autor bada.
5. Całość pracy wskazuje na dobrą znajomość autora w zakresie informatycznych narzędzi muzycznych (m.in. formatu MIDI, MATLABowego pakietu MIDI ToolBox, aplikacji SynthFont do syntezy kompozycji).

6. Należy pochwalić autora za próby różnych graficznych wizualizacji (w różnych miejscach pracy) obiektów, które z natury rzeczy dotyczą muzyki np.: wizualizacje macierzy przejść wzorców, wykresy pianolowe, wizualizacje macierzy kombinacji motywów, wizualizacje kompozycji molowych i durowych poprzez samoorganizujące się mapy, itp.

4.2 Główne uwagi krytyczne

1. Pewien niedosyt budzi brak jawnego zastosowania w algorytmach autora metod sztucznej inteligencji, które są deklarowane i w celu, i w tezie pracy. Algorytmy autora mają raczej charakter generatywny i częściowo transformacyjny, jak zauważa sam autor. Być może winowajcą w tym zakresie jest sam sposób sformułowania celu pracy (niepotrzebna wzmianka o metodach sztucznej inteligencji).
2. Z programistycznego punktu widzenia kody źródłowe autorstwa doktoranta są mało eleganckie lub miejscami wręcz niechlujne. Pojawia się w nich bardzo dużo wartości podanych „na sztywno”. Ma miejsce pomieszanie polskojęzycznych i anglojęzycznych nazw zmiennych. Nagminnym zjawiskiem są długie zestawy instrukcji `if` w miejscach, gdzie wystarczyłaby pętla po pewnej tablicy z zapisem jednej instrukcji `if`. Losując liczbę z rozkładu $(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$ autor posługuje się brzydkimi warunkami w stylu `if rand() < 0.33` lub `if rand() < 0.66` — czy te wartości są odpowiednimi przybliżeniami liczb $1.0/3.0$ i $2.0/3.0$ w typie `double`? Tego typu usterek programistycznych jest dużo.
3. Poza trzema wyróżnionymi procedurami algorytmicznymi (2.1, 2.2, 3.1) w pracy brakuje takich uporządkowanych zapisów. Autor zbyt często preferuje długi opis słowny nad bardziej zwartym zapisem algorytmicznym.

5 Uwagi i pytania stawiane na obronę rozprawy

Proszę autora o odniesienie się w trakcie publicznej obrony do następujących uwag i pytań.

1. Autor wspomina „muzyczną grę w kości” zaproponowaną przez Kirnbergera i Himmela (XVIII w.). Polega ona na składaniu kompozycji ze słownika gotowych fragmentów poprzez wskazywanie ich indeksów na podstawie sumy oczek wyrzuconych na dwóch kościach. Jak wiadomo, najczęstsza suma to 7 (prawdopodobieństwo $6/36$), potem 6 (prawdopodobieństwo $5/36$), itd. Pojawiają się tu ciekawe pytania. Chociażby: jak permutacja słownika zmienia styl wynikowych kompozycji?
2. Pomysły autora wymagają informacji wejściowej do algorytmów pochodzącej od człowieka (określenie motywu, wektorów z rytmiką, itp.), później już proces toczy się automatycznie z pewną losowością. Na ile trudne autor widzi zalgorytmizowanie tego pierwszego kroku? Czy udałoby się np. wygenerować automatycznie pewne parametry mini-generatorów, które autor nastawia ręcznie (tab. 4.1)?
3. W ramach ćwiczenia matematycznego, proszę doktoranta o przygotowanie stosownego slajdu na obronę z odpowiedzią na następujące pytanie. W łańcuchu Markowa pierwszego rzędu, ile wynosi prawdopodobieństwo pojawienia się w chwili t stanu j -ego pod warunkiem, że w chwili $t - 2$ system przebywa w stanie i -ym. Innymi słowy, ile wynosi: $P(q_t = S_j | q_{t-2} = S_i) = ?$
4. Na stronach 59, 60 autor używa współczynnika korelacji Pearsona do sprawdzenia, czy dany fragment muzyczny (jako wektor nut) x jest zgodny z pewną tonacją y wyrażoną

w 12-stopniowej skali. Autor stara się dalej wyjaśnić, jak na potrzeby wzoru (2.13) x i y zostają ujednolicone do tej samej długości. Prosiłbym o dokładniejsze wyjaśnienie podczas obrony.

5. Czy autor widzi jakiś związek proponowanej macierzy PTS z macierzą prawdopodobieństw przejść w modelach Markowa? Dlaczego akurat 40% najbardziej podobnych wzorców wchodzi do konkurencji podczas sklejania i dlaczego losowanie jednego z nich (spośród tych 40%) odbywa się zgodnie z rozkładem jednostajnym, a nie w pewien sposób proporcjonalny do podobieństw?
6. Algorytm 2.1 (str. 70) jest ogólnie zrozumiały, ale pojawiają się dwa pytania. Dlaczego oprócz wzorca melodii W_M autor generuje także wzorec linii basowej W_B , ale zwracane jest tylko W_M ? Czy zmienna położenia kolejnego dźwięku nie powinna być zwiększona o sumę $o + d$ (czyli przesunięcie + długość) nie zaś o samą wartość o ?
7. Ciekawym doświadczeniem byłoby przygotowanie dla słuchaczy na obronę utworu wygenerowanego sztucznie (muzyki tła) na bazie pewnej znanej rozpoznawanej powszechnie melodii. Byłbym zadowolony, gdyby doktorant zdołał przygotować taki utwór.

6 Pomniejsze uwagi

6.1 Usterki matematyczne, notacyjne, programistyczne

1. Mnożenia są zapisywane znakiem $*$ w tekście pracy (co dopuszczalne powinno być tylko w kodach źródłowych). Gwiazdki matematycznie kojarzą się raczej ze splotem (np. str. 14). Lepiej byłoby używać znaku $\cdot$ (w LaTeXu polecenie `cdot`) lub po prostu nie stawiać żadnego znaku.
2. Str. 17–20: Dość denerwujące są długie opisy słowne transformacji Schoenberga, czasami niezrozumiałe (np. transformacja ‘upside-down’). Zamiast podawać listingi na bieżąco lub krótkie formuły matematyczne autor zmusza czytelnika do wertowania pracy aż do spisu listingów.
3. Często w pracy używane są klamry $\{\dots\}$ przyjęte zwyczajowo dla zbiorów z zamiarem oznaczania wektorów (gdzie ważny dla autora jest porządek), zamiast nawiasów zwykłych lub kwadratowych.
4. Str. 21 (formalizm Xenakisa): Autor zużywa aż 6 linii tekstu żeby określić wektor v_m jako trójkę (p_n, d_n, i_n) tj. pitch, duration, intensity. Niezrozumiałe jest też w tym otoczeniu indeksowanie autora, np. $v_m = \{p_n, d_n, i_n\}$ — jaki jest związek m i n ? Chwilę później podane jest prawo dodawania wektorów zapisane jako:

$$(v_i = \{p_m, d_n, i_o\}) + (v_j = \{p_x, d_y, i_z\}) \Rightarrow (v_i + v_j) = \{(p_m + p_x), (d_n + d_y), (i_o + i_z)\}.$$

Notacja mocno rozdmuchana. Jaki sens wszystkich indeksów? Jaki sens strzałki $\Rightarrow$? Dużo prościej byłoby napisać po prostu: $(p_1, d_1, i_1) + (p_2, d_2, i_2) = (p_1 + p_2, d_1 + d_2, i_1 + i_2)$.

5. W wielu miejscach autor myli przedział liczbowy ze skończonym zbiorem. Np. na str. 21: „(...) wartości z przedziału 0–127 (...)”. Przypuszczalnie chodziło tu o zbiór $\{0, 1, \dots, 127\}$.
6. Str. 23: „(...) W przypadku gdy zbiór nie zawiera powtórzeń problem jest trywialny. Jednakże, gdy w zbiorze pojawiają się duplikacje (replikacje) danej nuty, rosną jej szanse na wylosowanie do kompozycji (...)”. Czy zbiór matematyczny może zawierać powtórzenia?

7. Str. 24: „(...) Suma wszystkich wartości prawdopodobieństwa wynosi 0”.
8. Str. 24, 25: Mówiąc o macierzy prawdopodobieństw przejść pomiędzy stanami (w modelu Markowa), autor używa niezręcznego określenia „macierz prawdopodobieństwa tranzycji” (liczba pojedyncza), a jeszcze innym razem mówiąc o tym samym „macierz transpozycji”.
9. Str. 161 (listing 5.8): Autor realizuje zliczanie na rzecz macierzy prawdopodobieństw przejść poprzez zagnieżdżone wielokrotne wywołania funkcji `find MATLABa`. W efekcie otrzymuje algorytm kwadratowy, podczas gdy wystarczyłoby liniowe przejście po macierzy nut.
10. Str. 35: Zaskakuje dziwnością algorytm genetyczny zapisany za pomocą diagramu UML maszyny stanów. Dlaczego autor nie posłużył się np. pseudokodem w formie ponumerowanych kroków? Co więcej w zapisie autora nie widać w sposób jasny iteracji głównej pętli genetycznej.
11. Wzór (2.1): Po co autor rozróżnia dwa przypadki odnośnie wartości ilorazu: długość wzorca / długość nuty? Przecież wystarczy tylko drugi przypadek $m_{MN} = \lfloor l_W / dur_W \rfloor$ — część całkowita tego ilorazu.
12. Str. 53: Warunek (2.4) wystarczy zapisać jako $m_i = m_j$. Jaki jest sens znaku równoważności $\Leftrightarrow$?
13. Str. 54, 55: Autor mógłby pomyśleć o szczególnym znaku dla operatora konkatenacji wzorców np. $\oplus$. W otoczeniu tych stron mieszane jest bowiem dodawanie zwykle z konkatenacją.
14. Str. 56: Wektor pozycji nut należało zapisać raczej jako $WPN = \{k_i : k_i = n + i \cdot rs, n \leq k_i \leq l\}$, gdzie $i = 0, 1, \dots$. Zapis autora jest niezrozumiały.
15. Większość lub mniejszość z równością jest w wielu miejscach pisana jako $\leq$ oraz $\geq$. Powinno być: $\leq$ oraz $\geq$.
16. Str. 57: Macierz kombinacji motywów lepiej zapisać jako zbiór wszystkich kombinacji np. w terminach autora jako $KMM = \{(i_1, i_2, \dots, i_k) : 1 \leq i_1 \leq i_2 \leq \dots \leq i_k \leq n\}$.
17. W wielu miejscach ma miejsce niejednorodność czcionki matematycznej w formułach tzw. wystawionych i w tekście. Np. na str. 57 mamy cc oraz cc , a także MM oraz MM .
18. Str. 63, 64: Autor miesza pojęcie miary z metryką. Dodatkowo, lepszym byłoby posługiwanie się oznaczeniem $\|\dots\|$ jako normy zamiast $|\dots|$.

6.2 Usterki edytorskie, literówki i inne

1. Str. 5: Autor deklaruje wyjaśnienia pojęć dotyczących dźwięku: częstotliwość, amplituda, tembr, czas trwania i obwiednia. Następują wyjaśnienia z pominięciem konkretnego wyjaśnienia pojęcia obwiedni.
2. Złe operowanie rodzajami (długością) myślników w całej pracy. Często autor stawia myślniki krótkie ‘-’ zamiast tzw. myślników pauz ‘—’ (np. str. 3, 5, 10).
3. Złe operowanie cudzysłowami. Autor stosuje zamerykanizowaną wersję “...” zamiast wersji przyjętej w języku polskim „...”. Np. na stronie 5 mamy “Glosariusz”, “sztuka generatywna”.

4. Str. 9: „wskazuję” zamiast „wskazują”.
5. Str. 10: podwójne stopniowanie przymiotnika: „(...) *bardziej ruchliwsza* (...)”.
6. Str. 10: przypuszczalnie brak słowa „jest” w zdaniu „*Ta cecha kompozycji* (...)”.
7. Str. 14: literówka „typo komunikaty” zamiast „typ komunikatu”.
8. Str. 14: gramatyka „*Sekwencja bajtów 4D 54 72 6B wyznaczają początek* (...)” — powinno być „wyznacza”.
9. Przy referencjach do numerów rysunków i tabel pojawiają się bardzo długie spacje pomiędzy skrótami „rys.” i „tab.” a samymi numerami. Przypuszczalnie autor nie stosuje w tym miejscu odpowiedniego symbolu w LaTeXu, przez co LaTeX traktuje napotkaną kropkę jako koniec zdania.
10. Źle sformatowana tabela 1.7.
11. Str. 26 (tab. 1.8): Niezgodność prawdopodobieństwa przejścia $28 \rightarrow 32$ pomiędzy tabelą (wartość 0) a tekstem (wartość 0.5).
12. Str. 26, 27: Niejednolitość oznaczenia miejsca dziesiętnego (np. 0,75 i 0.75).
13. Str. 28: Literówka w tytule pracy Chomsky’ego.
14. Str. 37: Błąd w nazwisku Stanisława Ulama.
15. Str. 37: „definiuję się” zamiast „definiuje się”.
16. Rys. 1.2.3: W otoczeniu tego rysunku recenzent nie znalazł wyjaśnienia skrótu GBMS nazwy własnej systemu autora. Można domyślać się jak skrót rozszyfrować patrząc na publikację [11].
17. Str. 40, 41: Podrozdział zatytułowany „Sztuczne sieci neuronowe generujące muzykę” nie zawiera informacji zgodnych z tytułem. Są tam ogólne informacje o falach EEG, odbiorze muzyki przez człowieka i bardzo ogólne informacje o sieciach neuronowych, ale brak jest podanych podejść do generowania muzyki nimi.
18. Str. 46: Nazwisko Markowa zapisane małą literą.
19. Str. 53: Zaskoczeniem są błędy językowe w zapisach autora „pół nuta”, „ćwierć nuta” — powinna być pisownia łączna.
20. Str. 56: Zamiast „Strukturę tą” powinno być „Strukturę tę” (biernik, a nie narzędnik).
21. Str. 57: Literówka w „(...) *generuje fragmentu muzyczne* (...)”.
22. Str. 64: Macierz PST tłumaczona jako *Pattern Transition Structure* (kolejność liter w skrótowcu).
23. Str. 76: „(...) *permutacje zostały skonstatowane* (...)” — powinno być raczej „skonkatenowane”.
24. Str. 114: Usterki edytorskie w nazwach parametrów: *onset*₄₂ i *rhythmic*_{value}.
25. Str. 132: Błąd ortograficzny „*cechą szczególnym*”.

7 Ocena dorobku naukowego autora

Cytowany w pracy dorobek autora obejmuje 5 publikacji: jedna wydana jako rozdział w monografii wydany przez Politechnikę Koszalińską, dwa artykuły w *Przeglądzie Elektrotechnicznym* oraz dwie publikacje konferencyjne. Na wyróżnienie zdecydowanie zasługuje jednoautorska publikacja [11] prezentowana przez Pana Mazurowskiego na konferencji ICNAAM 2014 (*International Conference on Numerical Analysis and Applied Mathematics*) na Rodos w Grecji. Jest to moim zdaniem wartościowa konferencja, a wydawane materiały są indeksowane w bazie Web of Science.

8 Konkluzja

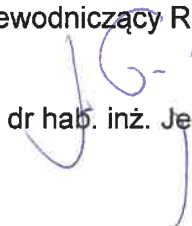
Pomimo wielu uwag krytycznych moja całościowa ocena rozprawy Pana Mazurowskiego jest pozytywna. Stwierdzam, że recenzowana praca spełnia wymagania określone przez ustawę z dnia 14 marca 2003 r. o stopniach naukowych i tytule naukowym (wraz z późniejszymi zmianami). Wnoszę o przyjęcie rozprawy i dopuszczenie mgr inż. Łukasza Mazurowskiego do publicznej obrony.

UCHWAŁA nr 71/2016/2017
Rady Wydziału Informatyki
Zachodniopomorskiego Uniwersytetu Technologicznego w Szczecinie
z dnia 4 lipca 2017 r.

**w sprawie nadania stopnia doktora nauk technicznych
w dyscyplinie Informatyka mgr inż. Łukaszowi Mazurowskiemu**

Zgodnie z Ustawą z dnia 14 marca 2003 r. o stopniach naukowych i tytule naukowym oraz o stopniach i tytule w zakresie sztuki (Dz. U. Nr 65, poz. 595 z późniejszymi zmianami) oraz ze zmianami w Ustawie z dnia 27.07.2005 r. *Prawo o szkolnictwie wyższym* Rada Wydziału Informatyki na posiedzeniu w dniu 4.07.2017r w wyniku głosowania tajnego podejmuje Uchwałę o nadaniu stopnia doktora nauk technicznych w dyscyplinie *Informatyka* mgr inż. Łukaszowi Mazurowskiemu.

Przewodniczący Rady Wydziału


dr hab. inż. Jerzy Pejaś