

Tworzenie narracji komputerowej gry fabularnej z użyciem transformacji grafowych

Streszczenie

Prezentowana rozprawa doktorska przedstawia formalny model opisu fabuły gry fabularnej i jego zastosowanie do wspomagania projektowania przygodowych gier wideo. Własności modelu przedstawione są na przykładzie prostej gry fabularnej. Opisana jest specyfikacja systemu StoryGraph 1.2 oraz referencyjna implementacja modelu, wykonana w języku Python.

Celem rozprawy jest opracowanie koncepcji opisu schematów fabularnych komputerowych gier przygodowych w oparciu o reprezentację grafową i grafową procedurę generacyjną. Przedstawione w rozprawie rozwiązanie zostało zweryfikowane praktycznie przez zaimplementowanie mechanizmów symulatora rozgrywki gracza gier komputerowych.

Zaproponowany w rozprawie nowy model grafowy wraz procedurą generacyjną jest odpowiedni do:

- opisu struktur fabularnych przygodowych gier wideo,
- implementacji mechanizmów symulatora rozgrywki gracza fabularnych gier przygodowych w celu testowania linii fabularnych,
- tworzenia mechanizmów współpracy projektowej w zespołowym tworzeniu gier,
- analizowania fabuł formalnie zakodowanych w postaci grafu rozgrywki, zarówno przeprowadzonej przez gracza w czasie rzeczywistym jak i hipotetycznej.

*

Praca składa się z sześciu rozdziałów i czterech dodatków.

W **rozdziale pierwszym** zaprezentowano kategorię gier przygodowych jako formę gry komputerowej na tle typologii gatunku oraz przedstawiono literaturoznawcze fundamenta teoretyczne, na których opiera się niniejsza praca, w szczególności prace Władimira Proppa, strukturalisty, którego teoria morfologii bajki magicznej zainspirowała nie tylko autorkę niniejszej rozprawy, ale także wielu innych badaczy struktur literackich na gruncie informatyki.

Rozdział drugi przedstawia praktyczny przykład gry, który będzie ilustracją tez zawartych w pracy i generatorem schematów fabularnych omawianych w pracy. Gra ta była podstawą twórczej pracy projektowej grup studentów w latach 2019–2022.

Rozdział trzeci zawiera formalne definicje matematyczne systemu grafowego będącego teoretyczną podstawą systemu. Wprowadzone jest tam pojęcie grafu n -warstwowego, grafu snopkowego (struktury grafowej stworzonej na potrzeby modelowania gier przygodowych) oraz dopasowania generycznego jako narzędzia określającego możliwy zakres modyfikacji i produkcji generycznej jako bezpośredniego odwzorowania akcji gracza.

Podstawą modelu fabuły jest graf zwany dalej grafem stanu świata gry, który opisuje aktualny stan świata i jest modyfikowany po każdym ruchu gracza i każdym działaniu postaci niezależnej. Graf ten składa się z wierzchołków odpowiadających obiektom świata gry i krawędzi reprezentujących relacje między tymi obiektami. Pod względem charakterystyki możemy wyróżnić cztery typy wierzchołków: odpowiadające lokacjom, bohaterom, przedmiotom oraz elementom wiedzy narracyjnej. Każdy typ wierzchołków wraz z relacjami między nimi tworzy warstwę.

Krawędzie tego grafu są skierowane. Ich znaczenie dla fabuły związane jest z tym, czy są krawędzie wewnętrzne reprezentujące relacje pomiędzy wierzchołkami z danej warstwy, czy krawędzie reprezentujące relacje pomiędzy wierzchołkami z różnych warstw.

Atrybuty traktujemy jako funkcje zdefiniowane na obiektach, których przeciwdziedziny określają zakres wartości zależny od własności obiektu, którą atrybut opisuje. Różne atrybuty określone są dla różnych typów wierzchołków warstw.

W celu optymalizacji wykonywania działań typowych dla mechanik gier wideo zaproponowano nową grafową procedurę generacyjną polegającą na transformowaniu grafów według formalnych reguł zwanych produkcjami. Implementacja zastosowania produkcji dla gier wideo wymaga użycia odpowiedniej struktury danych opartej o autorską koncepcję podgrafów *snopkowego* i *pólsnopkowego*.

Cały graf gry można traktować jako las podgrafów snopkowych osadzonych w podgrafie warstwy lokacji poprzez korzenie-lokacje. Wszystkie ścieżki wychodzące z danej lokacji rozgałęziają się, tworząc graf snopkowy. Prowadzi to do definicji grafu wielosnopkowego. Typowe modyfikacje świata wynikające z mechaniki gry najczęściej operują na jednym lub kilku snopkach.

Konsekwentne zastosowanie struktury snopków umożliwia zastosowanie specyficznego zagnieźdzonego zapisu grafu w formacie JSON, w którym przechowywane są informacje o grafie. Struktura snopkowa i umożliwiony przez nią zapis JSON, odbiegający od typowych reprezentacji grafowych struktur danych, zostały zaprojektowane pod kątem optymalizacji działań na grafie wynikających z mechanik gier fabularnych.

Podstawą rozgrywki są wydarzenia dziejące się w świecie gry. Typy możliwych wydarzeń narzuca mechanika gry. Aby można było mówić o modelu grafowym gry, każde wydarzenie należy opisać za pomocą produkcji grafowej procedury generacyjnej.

Warunki stosowalności. Nie zawsze samo istnienie podgrafu o określonej strukturze i mającego wierzchołki o określonych własnościach wystarczy do wykonania akcji. Takie dodatkowe ograniczenia znane są w literaturze jako warunki stosowalności albo predykaty stosowalności. Odwołują się one do wartości atrybutów w wybranym podgrafie i zwracają prawdę albo fałsz. Niespełnienie ich pozwala odrzucić produkcję mimo dopasowania lewej strony produkcji do grafu.

Instrukcje. W opisywanym modelu grafowym przedsięwzięcie akcji sprowadza się do wykonania ciągu instrukcji operujących na wierzchołkach wybranego podgrafu i strukturach snopkowych zbudowanych w oparciu o te wierzchołki.

Poziom ogólności. Produkcje możemy charakteryzować według różnych kryteriów, ale bardzo istotną cechą jest własność generyczności oparta o formalne mechanizmy pozwalające definiować akcje w sposób bardziej ogólny, tzn. nie ograniczając ich stosowania do węzłów o określonych etykietach.

Hierarchizacja akcji. Konsekwencją sformalizowania akcji gracza w postaci produkcji grafowych jest możliwość ich hierarchizacji, czyli wprowadzenia częściowego porządku w zbiorze produkcji. Taka hierarchia produkcji jest rozszerzeniem modelu formalnego o graf hierarchii produkcji. Można go traktować jako kolejny poziom opisu fabuły.

Rozdział czwarty i piąty jest przedstawieniem specyfikacji systemu informatycznego służącego do praktycznego wykorzystania zdefiniowanego modelu teoretycznego oraz analiza implementacyjna modelu, z uwzględnieniem opisu algorytmu dopasowania lewej strony produkcji do świata gry i możliwości jej aplikacji.

Na podstawie teoretycznego modelu grafowego zaproponowana została specyfikacja standardu implementacji StoryGraph, aktualnie dostępna jako StoryGraph 1.2. Zdefiniowane zostały niezbędne typy danych i wytyczne dotyczące ich szczegółowej implementacji. Standard implementacji jest elementem pośrednim między formalizmem matematycznym a kodem w konkretnym języku programowania. W oparciu o tę specyfikację działa zaimplementowany w języku Python system symulacji procesu decyzyjnego będący podstawą części praktycznej tej pracy.

Algorytmy stworzone na potrzeby implementacji to między innymi znajdowanie dopasowania generycznego grafu czterowarstwowego oraz budowa drzewa hierarchii produkcji. Pozwalają one zaprezentować skrypty umożliwiające:

- walidację danych,
- testowanie linii fabularnych (symulacja rozgrywki),
- modelowanie świata,
- wizualizację świata, produkcji oraz drzewa hierarchii produkcji.

Rozdział szósty przedstawia możliwe zastosowania systemu opartego o model grafowy w wielu aspektach: jego znaczenia dla tworzenia mechaniki gry, opartym na nim procesie projektowym, ze szczególnym uwzględnieniem projektowania zespołowego (ang. *collaborative design*), wykorzystywania mechanizmów grafowych do tworzenia gry w modelu bardziej lub mniej otwartego świata, analizy zachowań gracza a także oceny wpływu gry na schematy poznawcze gracza w grach poważnych (ang. *serious games*). Zasygnalizowana jest też możliwość wykorzystania mechanizmów systemu do analizy grywalności.

Wprowadzone jest pojęcie produkcji generycznych i szczegółowych oparte o wprowadzoną w modelu funkcję pełnego i częściowego etykietowania i ich rozszerzenie w postaci struktury produkcji pochodnych w drzewie hierarchii. Jest to możliwe dzięki wprowadzeniu częściowego uporządkowania produkcji.

System, w którym tylko produkcje szczegółowe są dostępne dla gracza, powoduje, że gra jest zamknięta, tzn. możliwe są tylko interakcje między wskazanymi *explicite* elementami świata gry. System, w którym na podstawie produkcji generycznych umożliwiane są interakcje gracza z dowolnymi elementami świata, jest znacznie bardziej otwarty, ale zwiększa się niebezpieczeństwo, że gracz znajdzie drogę „na skróty”, albo możliwe działania będą niezgodne z fabułą lub zdrowym rozsądkiem. W pracy przedstawione zostały przykładowe rozwiązania jednego i drugiego typu z omówieniem różnic struktur je opisujących.

Opisany został także oparty o przedstawiony model proces zespołowego projektowania gry wideo a także wymienione praktyczne projekty wykorzystujące ten model, zarówno prace studenckie jak i projekt badawczy. Na końcu przedstawione zostały perspektywy rozwoju systemu.