

Michał Turek\*

## Zmodyfikowane gramatyki kształtu w zastosowaniu do generowania siatek 3D

### 1. Wprowadzenie

Gramatyki kształtu znajdują zastosowania przy przetwarzaniu różnego rodzaju obiektów, które można reprezentować tzw. kształtem. Kształt może być naturalnym kształtem geometrycznym, może także być określony jako dowolnie określony sygnał dający się zidentyfikować. W ogólnym zarysie gramatyka kształtu zadana jest jako czwórka

$$G = \langle S, L, P, I \rangle,$$

gdzie:

- S – zbiór kształtów,
- L – zbiór symboli,
- P – zbiór produkcji (reguł kształtu),
- I – zbiór początkowy.

Przetwarzanie z udziałem gramatyki, czyli generowanie nowego kształtu polega na rozpoznaniu podkształtu i zastąpieniu go innym poprzez zastosowanie wybranej z P reguły kształtu. Reguła taka definiuje owo podstawienie.

Klasyczne gramatyki kształtu zawierają pulę homogenicznych reguł kształtu (P), przenoszących informacje o możliwych tranzycjach kształtu – przetwarzanego przez gramatykę. Takie podejście, choć niezmiernie elastyczne, nie znajduje licznych zastosowań w praktyce [5]. Głównym problemem jest brak sprawnego algorytmu weryfikującego reguły i wyrokującego o możliwości zastosowania konkretnej z nich w zadanej sytuacji. Przy wąskiej puli symboli w L deterministyczne dopasowanie kształtu (np. pochodzącego od naturalnego obiektu) do symbolu gramatyki jest praktycznie niemożliwe. Algorytm dopasowania, określający swoją treścią procedurę dopasowania, tzw. *Left Hand Side* reguł (o którym niżej) do kształtów obrabianych, stanowi podstawę dla podejmowania decyzji o wyborze reguł kształtu [4]. Gdy informacja, na jakiej bazuje, jest szczątkowa – powstaje zbyt wiele możliwości dokonania tranzycji danego kształtu, pomiędzy którymi trzeba wybierać. Błędny wybór (dokonywany często na podstawie wartości przypadkowych parametrów lub wręcz losowo) spowoduje wygenerowanie kształtu niezgodnego z oczekiwaniami.

---

\* Katedra Automatyki, Akademia Górniczo-Hutnicza w Krakowie

Ominięcie problemu niedeterministycznego wyboru reguł gramatyki kształtu będzie wymagało rozwiązań zakładających rozbudowę gramatyk kształtu o nowe struktury. Klasyczne gramatyki kształtu dopuszczają możliwość tworzenia kształtów złożonych jedynie w oparciu o dodatkową wiedzę na temat położenia figur (np. przyleganie) [2] – to jednak ciągle zbyt mało. Zaproponowane w artykule rozwiązanie rozszerza definicje wiedzy dodatkowej, określanej gramatyką. Zostało ustabilizowane w konsekwencji licznych prób dopasowywania systemu klasyfikacji reguł kształtu do potrzeb faktycznych przekształceń obiektów 3D. Obiekty te, funkcjonując w świecie rzeczywistym, stanowiły wzorce dla transformacji i były analizowane zarówno pod kątem przestrzennej konstrukcji, jak i przynależności funkcjonalnej. Umożliwiło to opracowanie wciąż elastycznego modelu i dodatkowo nie abstrahującego od dziedzin, z którymi przetwarzane obiekty są powiązane (np. architektura danego budynku, model symulacyjny zadanego procesu itp.).

## 2. Założenia wstępne

Przetwarzanie siatki 3D w gramatyce kształtu rozpoczyna się od analizy tzw. kształtu początkowego. Kształt ten jest podawany jako parametr planowanej transformacji. Definicja takiego kształtu, zgodnie z teorią, może być dość dowolna. Kształt można przykładowo zakodować jako siatkę łączonych odcinkami wierzchołków, gotowych wielokątów, jako zbiór zdefiniowanych na potrzeby gramatyki symboli lub wręcz jako dowolny inny sygnał – będący rozpoznawalną porcją informacji. Same gramatyki kształtu (*Shape feature description language*) [8] wywodzą się z tematyki geometrii analitycznej. Tam poprzez zastosowanie zapisu formalnego pojawia się możliwość definiowania wszelkiego typu obiektów (docelowo także nie-geometrycznych) oraz określania transformacji, jakim te obiekty mogą podlegać. Definicja gramatyki kształtu nie określa reguł takich transformacji, nie narzuca też dziedzin w jakiej gramatyka kształtu mogłaby znaleźć zastosowanie. Intuicyjnie przyjmuje się, iż będą to dane reprezentujące bryły w przestrzeni  $n$ -wymiarowej. Okazuje się jednak, że dane te można zastąpić definicjami zupełnie innych encji, będących przykładowo sygnałem dźwiękowym, elektrycznym czy obrazem rastrowym. Jedyнным warunkiem jest tu opracowanie algorytmu dopasowującego wspomniany kształt do elementów zbioru  $S$  gramatyki. Gramatyki kształtu zostały po raz pierwszy formalnie opisane w roku 1972 przez Stiny'ego i Gipsa [6]. Przez ponad trzydzieści lat koncepcja gramatyk kształtu była wyjściem dla szeregu różnorodnych opracowań, mających na celu udoskonalenia lub utylizację przetwarzania kształtów w zastosowaniu praktycznym. Początkowo gramatyka kształtu składała się z trzech elementów [7]:

- 1) kształtu początkowego,
- 2) zbioru kształtów,
- 3) zbiór reguł kształtu.

Każda reguła posiadała dwa elementy kształt postawiony z jej lewej strony (tzw. LHS – *Left-Hand Side shape*) oraz kształt postawiony z jej prawej strony (tzw. RHS – *Right-Hand Side shape*). Regułę można więc zaprezentować graficznie jako: LHS  $\rightarrow$  RHS. Reguła taka może przeprowadzić kształt (lub jego fragment) dopasowany do LHS w nowy (rys. 1). Algorytm sterujący przetwarzaniem kształtu dokonuje iteracyjnego „odpalania” dopasowanych reguł kształtu. Dopasowanie reguły nastąpi wówczas, gdy LHS będzie pasował do fragmentu kształtu przetwarzanego. Słowo „pasował” budzi tu pierwsze wątpliwości, gdyż jednoznaczne wyrokowanie o dopasowaniu kształtu nie jest problemem banalnym. Często [6] zakłada się, iż kształt musi być identyczny z LHS pod kątem proporcji odległości wierzchołków składowych [9] – w rozumieniu metryki euklidesowej. W celu uzyskania tej „identyczności” dozwolone są jedynie podstawowe transformacje modyfikujące kształt (głównie skalowanie).



Rys. 1. Funkcjonowanie reguły kształtu przy prostym dopasowaniu LHS i fragmentu kształtu

Jak wspomniano we wprowadzeniu, podstawowym założeniem stało się uzależnienie projektowanego modelu reguł od relacji występujących pomiędzy obiektami w świecie rzeczywistym. Reguły kształtu będą zawierały transformacje, przeznaczone do modyfikowania właśnie takich obiektów. Wprowadzenie analogii do rzeczywistego świata obiektów 3D jest więc w pełni uzasadnione.

### 3. Klasy reguł: relacja *gen-spec*

Podstawowym problemem, napotykanym przy analizie LHS reguł jest brak dostatecznie uporządkowanych danych, mogących stanowić podstawę do orzekania o zgodności kształtu i LHS reguły. Pozostawienie puli reguł w jednym nieklasyfikowanym dalej zbiorze nie jest tu środkiem wystarczającym. Koncepcja wprowadzenia klas reguł ma na celu opracowanie mechanizmów dodatkowej fragmentacji tego zbioru, dostarczając tych brakujących danych. Jej szkic jest banalny i wywodzi się z powszechnie znanego hierarchicznego modelu klas i instancji klas, stosowanego choćby w obiektowych językach programowania. Klasa reguł zyska wówczas swoje przeznaczenie – jako uprawniona do modyfikacji konkretnego typu kształtów 3D – charakteryzującego się konkretnymi atrybutami i niekiedy

także konkretnymi wartościami atrybutów. Pula takich atrybutów jest otwarta – z możliwością jej modyfikacji (a szczególnie rozszerzania) podczas wyprowadzania jednej klasy reguł kształtu z innej. Klasa reguł powstaje zatem na wzór innej klasy, co jest typowym przykładem dziedziczenia atrybutów w modelu klas. Powstająca w konsekwencji nadklasa reguł i podklasa reguł (wyprowadzona) będą powiązane relacją typu *gen-spec*. Omawiane atrybuty będą elementami siatek 3D (kształtów), dopasowywanymi za pomocą zdefiniowanych metryk – jednak już w sposób porządkowany, dając podstawę do także uporządkowanej klasyfikacji dopasowania LHS nowej reguły kształtu do rozważanego aktualnie kształtu przetwarzanego przez gramatykę. Powyższe założenia postulują dodanie do definicji gramatyk kształtu zbioru C klas reguł kształtu oraz zbioru RC – relacji typu nadklasa-podklasa pomiędzy elementami zbioru C. Każdy element zbioru produkcji P (reguły kształtu) będzie powiązany z elementem zbioru C (każda reguła kształtu będzie miała swoją klasę).

#### 4. Agregacja reguł: relacja *whole-part*

W uzupełnieniu relacji *gen-spec* nasuwa się oczywista sugestia – wprowadzenie idącej w parze z *gen-spec* relacji fizycznej przynależności. Tym samym *gen-spec* umożliwi określanie specjalizacji i przynależności gatunkowej obiektów 3D, zaś druga relacja – ich fizyczną przynależność do kształtu złożonego. Obydwie relacje będą się funkcjonalnie uzupełniały, niosąc dalszą informację dla algorytmu sterującego odpalaniem reguł kształtu. Nie trudno się domyślić, iż aby system zbudowany na takiej koncepcji działał sprawnie, to relacje łączące reguły powinny istnieć także między samymi kształtami. Istnienie relacji *whole-part* będzie warunkowało wykorzystanie danej reguły kształtu w sposób następujący:

„Jeżeli LHS określa kształt A będący częścią lub częściami innego kształtu B oraz kształt wzorcowy jest częścią lub częściami kształtu B, to następuje uzgodnienie LHS reguły.”

W takiej sytuacji algorytm walidujący przed odpaleniem reguły nie będzie bazował jedynie na metryce podobieństwa kształtu i LHS (zbiór S), ale przede wszystkim na zgodności klasy LHS reguły i klasy kształtu (przeznaczenie kształtu) oraz zgodności agregacji reguły i kształtu (przynależność kształtu do innych). Założenie postawione w bieżącym podpunkcie postuluje dodanie do definicji gramatyk kształtu także zbioru A relacji zawierania pomiędzy elementami zbioru kształtów S.

Reasumując: zmodyfikowana gramatyka kształtu to siedem zbiorów:

$$G_m = \langle S, L, P, I, C, RC, A \rangle,$$

gdzie:

- S – zbiór kształtów,
- L – zbiór symboli,
- P – zbiór produkcji (reguł kształtu),
- I – zbiór początkowy,

- C – zbiór klas reguł kształtu,
- RC – zbiór relacji typu nadklasa-podklasa,
- A – zbiór relacji zawierania.

## 5. Implementacja i testy

Zarysowana w poprzednich podrozdziałach koncepcja wymaga naturalnie dogłębnej weryfikacji praktycznej. Trudno tu o formalny dowód poprawności metody, gdyż zalety jej funkcjonowania widoczne są dopiero przy ocenie funkcjonowania konkretnych gramatyk przetwarzających kształty postrzegane przez człowieka – czyli dopiero przy zastosowaniu konkretnego zbioru kształtów  $S$ , oraz uzupełniających go relacji. Największe problemy implementacyjne spodziewane były przy definiowaniu i kodowaniu treści metryki dopasowania LHS reguły do kształtu. Okazało się jednak, iż w przypadku istnienia dodatkowych relacji pomiędzy kształtami i regułami znaczenie metryki geometrycznego dopasowania zostało kształtów usunięte na dalszy plan. Kluczową przesłanką dla weryfikacji reguł i ich odpalania stała się przynależność do danej klasy lub zagregowanie w kształcie .

Pierwszy z przeprowadzonych testów dotyczył przetwarzania abstrakcyjnej przestrzeni symboli graficznych. Wyróżnikiem klas symboli był tu kolor z rozbięciem na poszczególne składowe (R,G,B) oraz liczba i ułożenie określających je wierzchołków. Przy odgórnie zdefiniowanym modelu klas reguł (14 reguł kształtu, 4 klasy, 4 relacje typu nadklasa-podklasa, 2 relacje agregacji wiążące łącznie 12 reguł) testowano algorytm na losowo podawanych siatkach 3D (jako kształt początkowy), także losowo przypisywanych do klas. Kolejność odpalania reguł [1] w każdym przypadku była w pełni przewidywalna. Określenie klasy kształtu początkowego przy tej ilości reguł także było wystarczające (nie było wymagane definiowanie relacji *whole-part* dla kształtu początkowego).

Dalsze testy prowadzone były na zbiorach zawierających od 100 do 300 elementów, zaś wyniki monitorowane wizualnie. Rozbudowana gramatyka, definiowana tymi zbiorami, była przeznaczona do generowania budynków wielopiętrowych. Reguły – dzięki stosowaniu relacji *gen-spec* – mogły być produkcjami tworzącymi zarówno ogólny kształt budynku jak i detale architektoniczne. Jako kształt początkowy zastosowano plan 2D przyszłego budynku, zaś klasa tego kształtu określała wysokość liczoną w kondygnacjach. Produkty wytworzone z użyciem gramatyki były zaskakująco poprawne, co pozwala myśleć bardzo optymistycznie o jej przydatności [3].

## 6. Podsumowanie

Jak rozważano – przeprowadzenie dowodu formalnego poprawności i użyteczności metody nie jest możliwe. Testy praktyczne wykazały jednak, iż w konkretnej rzeczywistej sytuacji zaprojektowanie automatu przetwarzającego kształty także na bazie ich przynależności do klas oraz relacji zawierania jest dużo łatwiejsze. Daje ponadto bardziej przewidy-

walne rezultaty. Przygotowana przez autora implementacja metody umożliwia zbudowanie bazy kształtów, reguł kształtu oraz siatki relacji pomiędzy tymi obiektami. Baza taka może być podstawą do praktycznego wykorzystania metody – mogącej z powodzeniem automatyzować prace projektowe w dziedzinie modelowania kształtów, wspomagać czynności projektowe w zastosowaniach architektonicznych czy różnorodnych obliczeniach symulacyjnych.

## Literatura

- [1] Foley J.D., van Dam A., *Wprowadzenie do Grafiki Komputerowej*. WNT, 1995.
- [2] ShangChing Chou, *A Rule Based Geometry Theorem Prover Using Fullangles*. The Wichita State University Press, 2004.
- [3] Buss S.R., *A Mathematical introduction to OpenGL*. Cambridge University Press, 2003.
- [4] Han J., *Characteristic Rules*. PAN.
- [5] Mulawka J., *Systemy Ekspertow*. PAN, 1996.
- [6] Stiny G., Gips J., *Shape grammars and the generative specification of painting and sculpture*. Information Processing 71 Ed. C.V.Freiman (North-Holland, Amsterdam), 1972, 1460–1465.
- [7] Stiny G., *Introduction to shape and shape grammars*. Environment and Planning B: Planning and Design, vol. 7, 1980, 343–351.
- [8] Dassow J., *Grammars with Regulated Rewriting*. University of Magdeburg, 2006.
- [9] Knight T., *Shape Grammars in Education and Practice: History Aspects*. MIT 2000.