

METODY WSPOMAGANIA PROJEKTOWANIA ARCHITEKTURY MIESZKANIOWEJ - OD METOD SYSTEMOWYCH DO GRAMATYKI FORM

Aleksander Asanowicz

Wydział Architektury, Politechnika Białostocka, ul. Grunwaldzka 11/15, 15-893 Białystok
E-mail: asan@pb.edu.pl

METHODS SUPPORTING THE DESIGN OF HOUSES - FROM SYSTEMATIC APPROACH TO SHAPE GRAMMAR

Abstract:

In the paper two methods supporting design of houses will be considered. The first method was created on the basis of a systematic approach, especially the soft systems methodology. This approach abandoned the deterministic way of designing and the deterministic design strategies. Soft methodology considered the design process as a dialog. The second approach is based on a mathematics modeling method - shape grammar. Shape grammar connects spatial and quantitative aspects of the form. Presented in this paper method based on the description grammar, which concerns the description of the quality of the form, and shape grammar, which is used for the plan generation.

Streszczenie:

W referacie przedstawiona zostanie ewolucja metod wspomaganie projektowania architektury mieszkaniowej, poczynając od metod bazujących na podejściu systemowym, a kończąc na współczesnych metodach modelowania matematycznego. Przedstawione będą dwa programy umożliwiające twórcze poszukiwanie i generowanie idei projektowych. Pierwszy powstał w nurcie „miękkiego” podejścia systemowego, drugi zaś w oparciu o gramatykę form. „Miękkie” podejście systemowe (*soft systems methodology*) odrzucało podejście deterministyczne w projektowaniu architektonicznym i rezygnowało z opartych na tym podejściu strategii. Uwzględniło ono dialogowy charakter procesu projektowania. Gramatyka form łączy przestrzenne i ilościowe aspekty form. Prezentowana metoda bazuje na zastosowaniu zasad gramatyki deskryptywnej (*description grammar*) i gramatyki form (*shape grammar*). Gramatyka form tworzy rzut obiektu, a reguły deskryptywne odnoszą się do opisu funkcji.

Keywords: systematic methods, CAD, shape grammar

Słowa kluczowe: metody systemowe, komputerowe wspomaganie projektowania, gramatyka form

WPROWADZENIE

Projektowanie budynków mieszkalnych dostępnych dla szerokiej grupy ludności stało się ważnym problemem od początku rewolucji przemysłowej w XVIII wieku, w związku z masowym napływem ludności ze wsi do miasta. Wszyscy wielcy dwudziestowieczni architekci podejmowali próby rozwiązania tego problemu. Jako przykład podać można propozycje W. Gropiusa zabudowy mieszkaniowej w Törten (1928), Le Corbusiera w Pessac (1929) czy F.L. Wrighta - Usonian houses (1954). Wspólną cechą tych projektów było dążenie do opracowania schematu umożliwiającego generowanie różnorodnych rzutów domów dostępnych i pozwalającego na jego

przemysłową realizację przy minimalizacji nakładów finansowych.

Dążenie do stworzenia uniwersalnego systemu projektowego jest zgodne z odwiecznym dążeniem człowieka do racjonalizacji swoich działań.

Dążenie do ułatwienia sobie pracy i równocześnie do osiągania coraz wyższej jakości tworzonych dzieł towarzyszy człowiekowi od początku procesu ewolucji. Wraz z komplikowaniem się tworzonych przez niego obiektów komplikowały się również narzędzia i metody, których używał. Wiek XX przyniósł tak znaczące zmiany we wszystkich dziedzinach życia, że metody proste, bazujące na ewolucyjnych zmianach metod i środków nie były w stanie sprostać

potrzebom. Powstała metodologia projektowania, nauka, celem której było wyjaśnienie i projektowanie procesu projektowania. Jednym z jej znaczących nurtów były metody projektowania bazujące na teorii systemów, cybernetyce i wykorzystaniu aparatu matematyczno-logicznego.

1. METODY SYSTEMOWE W PROJEKTOWANIU

Metody systemowe zawierały w sobie metody obiektywizacji procesów oceny i wyboru rozwiązań wariantowych, metody optymalizacji działań projektowych i modelowanie oparte na rachunku prawdopodobieństwa. W latach 60-tych XX wieku podjęto próby stworzenia skomplikowanych systemów automatyzacji projektowania. Były to systemy pierwszej generacji, bazujące na tak zwanym „twardym” podejściu systemowym. Metody te były stosowane w programowaniu i projektowaniu architektonicznym do przeprowadzania analiz ekonomicznych, obliczeń konstrukcyjnych, planowania procesu projektowania i sterowania jego przebiegiem. Były również podejmowane pierwsze próby automatyzacji opracowywania dokumentacji projektowej, szczególnie w projektowaniu obiektów typowych.

Oczywistość i celowość stosowania podejścia systemowego w projektowaniu architektonicznym nie budziła żadnych wątpliwości. Największe nadzieje wiązano z metodami funkcjonalnego rozplanowania obiektów (zakłady przemysłowe, szpitale, szkoły, wielorodzinne budynki mieszkalne). Wybór takich obiektów wynikał z możliwości ilościowego przedstawienia przebiegających w tych obiektach procesów funkcjonalnych.

Praktyka projektowa wykazała jednak niewielką przydatność metod systemowych pierwszego pokolenia. Było to wywołane głównie przez skomplikowanie i pracochłonność etapów przygotowawczych. Projektant, mając do dyspozycji metody komputerowego wspomaganie projektowania, doskonale z technicznego punktu widzenia, popełniał takie same błędy, ale dodatkowo popełniał je w większej skali. Metody systemowe sprawdziły się jedynie w projektowaniu infrastruktury technicznej i obliczeniach konstrukcyjnych. Efekty w dziedzinie architektury były bardzo wątpliwe, szczególnie w zakresie działań optymalizacyjnych. Zupełnie nieadekwatne do natury projektowania okazały się matematyczne metody optymalizacji, statystyka, kwilimetria. Najpełniej nieadekwatność ta przejawiała się w pracy architektów i urbanistów.

Na bazie krytyki metod pierwszej generacji sformułowano zasady „miękkiego” podejścia systemowego (*soft systems methodology*), które odrzucało podejście deterministyczne w projektowaniu architektonicznym i rezygnowało z opartych na tym podejściu strategii. Nowe metody uwzględniały dialogowy charakter procesu projektowania (Rittel, 1972). Były one stosowane w analizie przepływu informacji projektowej, procesu podejmowania decyzji, ich weryfikacji i oceny. Do najbardziej znanych metod drugiej generacji możemy zaliczyć takie metody, jak: HIDECS-3 (Alexander, 1963), Pattern Language (Alexander, 1964) i Flatwriter (Friedman, 1971).

2. FLATWRITER

Flatwriter, autorstwa Y. Friedmana, miał służyć do projektowania struktur mieszkalnych. Flatwriter to wspomagana komputerowo metoda, opierająca się na udziale mieszkańców w projektowaniu własnych mieszkań. Było to narzędzie umożliwiające przekazanie użytkownikowi informacji o projektowanym dla niego obiekcie. Program ten pozwalał na indywidualny wybór spośród dużej ilości rozwiązań kombinatorycznych oraz na przeprowadzenie korekty niewłaściwych decyzji bez udziału architekta. Wykorzystując opracowaną przez Friedmana metodę, użytkownik sam określał położenie i układ funkcjonalny swego mieszkania. Równocześnie otrzymywał od systemu informację o skutkach podejmowanych przez siebie decyzji. Projektowanie w programie Flatwriter odbywało się za pomocą specjalnie na potrzeby tego programu zaprojektowanego oprzyrządowania:

- Klawiatura I - zawierała 53 klawisze, na których były przedstawione możliwe konfiguracje mieszkania, pokoi, węzłów sanitarnych, kuchni. Za jej pomocą można było stworzyć 240 milionów konfiguracji układów funkcjonalnych.
- Klawiatura II - służyła do zapisu przyzwyczajzeń użytkownika, dotyczących codziennych funkcji, np. częstotliwości korzystania z danych pomieszczeń.
- Drukarka - służyła do wydruku planów i informacji o nakładach finansowych, komforcie, zużyciu energii w czasie eksploatacji danego wariantu mieszkania.
- Monitor I - do wyświetlania danych o dostępności miejskiej infrastruktury.
- Monitor II - do wyświetlania w czasie rzeczywistym informacji o skutkach podejmowanych działań.

Projektowanie za pomocą Flatwritera można podzielić na osiem etapów:

Etap I

Korzystając z Klawiatury I, dokonywano wyboru konfiguracji (rozplanowanie mieszkania, określenie formy poszczególnych pomieszczeń, położenie kuchni i węzłów sanitarnych, i określano kierunek północy. Następnie wykonywany był wydruk planu mieszkania.

Etap II

Sprawdzenie zgodności rozwiązania funkcjonalnego z wymaganiami użytkownika. Jeśli rozwiązanie nie zadowalało użytkownika, to powtarzano działania I etapu.

Etap III

Opracowanie kosztorysu wybranego rozwiązania funkcjonalnego i jego wydruk.

Etap IV

Sprawdzenie zgodności finansowych możliwości użytkownika z kosztorysem.

Jeśli rozwiązanie okazywało się nieodpowiednie (zbyt drogie lub zbyt tanie), to powtarzano działania I etapu. Jeśli użytkownik akceptował kosztorys, to korzystając z Klawiatury II, wprowadzał informacje o częstotliwości wykonywania codziennych funkcji i definiował prognozę ich wykonywania w nowych warunkach.

Etap V

Analiza dokonanych wyborów z punktu widzenia zwyczajów użytkownika. Efektem analizy była między innymi prezentacja informacji o zużyciu energii w wyniku realizacji przyzwyczajzeń mieszkańców. Jeśli przedstawione dane odpowiadały użytkownikowi, to przechodzono do etapu następnego. Jeśli nie, to możliwy był powrót do etapu I lub IV i ponowna analiza stylu życia.

Etap VI

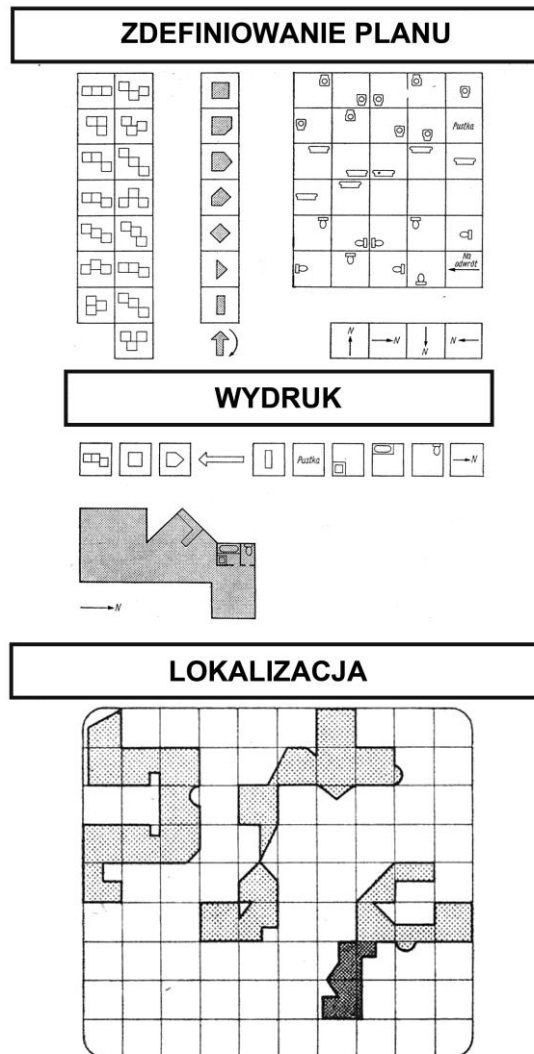
Korzystając z informacji przedstawionej na Monitorze I (rozmszczenie i dostępność sieci energetycznej, wodociągowej i kanalizacyjnej), przeprowadzano wybór lokalizacji mieszkania w ramach struktury mieszkaniowej.

Etap VII

Sprawdzenie rozwiązania w dwóch aspektach. Pierwszy - czy wybór indywidualny nie jest sprzeczny z wyborami innych użytkowników. Drugi aspekt wiązał się z wydolnością infrastruktury struktury mieszkaniowej. Jeśli analiza nie wykryła żadnych sprzeczności, to przechodzono do etapu ostatniego. W innym przypadku powtarzane były działania etapu VI.

Etap VIII

Wydruk projektu, kosztorysu i analiza skutków dokonanych przez użytkownika wyborów.



Ryc. 1. Elementy systemu Flatwriter

Przedstawiona powyżej metoda projektowania najlepiej odzwierciedla tendencje wspomaganie projektowania poprzez wdrożenie programów opartych na zasadach „miękkiego” podejścia systemowego.

Metody systemowe mimo swych oczywistych zalet nie zostały powszechnie wdrożone. Powodem były tak trudności techniczne (brak komputerów odpowiedniej mocy i pakietowy model przetwarzania danych), jak i opór mentalny projektantów, nieprzygotowanych do zmiany tradycyjnego sposobu projektowania. W ostatnich latach obserwujemy renesans podejścia systemowego w projektowaniu architektonicznym. Jest to rezultatem rozwoju informatyki z jednej strony, z drugiej zaś powszechnej akceptacji projektowania wspomaganego komputerowo.

3. METODY MATEMATYCZNE W PROJEKTOWANIU ARCHITEKTONICZNYM

Obecnie szczególnie intensywnie rozwijają się badania metod projektowania opartych na modelowaniu matematycznym. Programowanie za pomocą metod matematycznych okazuje się nadzwyczaj efektywne nie tylko w projektowaniu układów funkcjonalnych, ale również w poszukiwaniu i modelowaniu form przestrzennych. Metody matematyczne mogą wydawać się trudne do opanowania przez niematematyków, a w szczególności trudne do zaakceptowania przez architektów, tak jak to miało miejsce w przypadku metod systemowych. Na szczęście powstało szereg instrumentów, które ułatwiają proces programowania, a co za tym idzie, czynią go dostępnym dla niematematyków. Jedną z takich metod jest gramatyka form (*shape grammar*). Ma ona nie długą historię. Pierwsza praca poświęcona gramatyce form została opublikowana dopiero w roku 1972 (Stiny G., Gips J., 1972). W projektowaniu architektonicznym określenie „gramatyka” było najczęściej używane metaforycznie. W pracy G. Stiny’ego, gramatyka jest rozpatrywana jako matematycznie sformalizowany sposób przekształceń figur geometrycznych. Jest ona stosowana do generowania możliwych konfiguracji w ramach określonej gramatyki.

Mimo że zwykle uważa się, iż obliczenia odnoszą się do sfery liczb, to gramatyka form opiera się na połączeniu charakterystyk przestrzennych i ilościowych. Jest to podejście całkowicie nowe, gdyż tradycyjnie ludzie są przyzwyczajeni do rozpatrywania tych dwóch parametrów oddzielnie.

Gramatyka form składa się z form, znaków - markerów, stosunków przestrzennych, reguł odnoszących się do form i formy matki. Forma to dowolny zakończony układ punktów, linii, powierzchni i brył. Jest ona elementarną składową gramatyki. Stosunki przestrzenne odnoszą się do stosunków między formami, czyli inaczej mówiąc, określają organizację form. Reguły to instrukcje, które są stosowane rekurencyjnie do generowania stosunków przestrzennych. Reguła składa się z dwóch części połączonych strzałką. Strzałka informuje, że forma po lewej stronie jest przekształcana w formę po stronie prawej (bądź jest przez nią zamieniana). Sformalizowany zapis gramatyki można przedstawić następująco:

$$A \rightarrow A = B \text{ i } B \rightarrow B = A,$$

gdzie A i B oznaczają formy, A + B stosunki przestrzenne między formami, a strzałka przedstawia regułę.

Reguły gramatyki określają, w jaki sposób zgodnie z przyjętymi stosunkami przestrzennymi możemy dodać formę. Ponadto różne przekształcenia przestrzenne (obrót, przemieszczenie, odbicie, skalowanie) definiują różnorodne sposoby zastosowania danej reguły. Reguły są stosowane w deterministycznej kolejności od pierwszej do ostatniej, a następnie proces ten jest powtarzany.

Reguły określają gramatykę form, a gramatyka form z kolei określa styl, który możemy traktować jako pewnego rodzaju przestrzeń projektowania - macierz wszelkich możliwych obiektów, które mogą być stworzone w ramach danej gramatyki. Rozmiar przestrzeni projektowania jest definiowany poprzez ilość kroków i jest wyliczany za pomocą prostego wzoru kombinatorycznego:

$$D = L^N,$$

gdzie D - ilość możliwych projektów, które powstają w rezultacie wykonania N kroków, a L - liczba pozycji markerów (inaczej mówiąc, reguły), które muszą być zastosowane w każdym kroku.

W ten sposób liczba rozwiązań zależy od tego, ile razy była stosowana reguła gramatyki. Obliczenia są kontynuowane rekursywnie do momentu uzyskania zadowalającego rozwiązania. W prostych projektach z 4 krokami otrzymujemy 256 rozwiązań, a gdy liczba kroków wzrasta do 8, to uzyskujemy 65 tysięcy rozwiązań.

Początkowo gramatyka form była stosowana do analizy. Pierwsze analityczne zastosowanie gramatyki przedstawił G. Stiny w 1977 roku (Stiny, 1977). Praca ta ustanowiła standardy dla następnych badań w zakresie gramatyki form. Praca Stiny’ego zalicza się do gramatyk parametrycznych. Wykorzystując pięć prostych reguł, gramatyka opisuje zasady kompozycji istniejących krat (siatek) i tworzy nieskończoną ilość nowych, hipotetycznych projektów w tym samym stylu. Drugie analityczne zastosowanie gramatyki form przedstawili Stiny i Mitchell w 1978 roku. Praca dotyczyła gramatyki obiektów A. Palladia (Stiny, Mitchell, 1978).

4. GRAMATYKA FORM DLA OSIEDLA MALAQUEIRA

Jedną z najciekawszych współczesnych prac poświęconych zastosowaniu gramatyki form w projektowaniu architektonicznym jest praca J.P. Duarte, który stworzył gramatykę dla osiedla Malagueira, projektowanego od 1978 roku przez A. Sizę (Duarte, 2001)

A. Siza opracował schemat, który umożliwił generowanie budynków mieszkaniowych różnego ty-

pu. Dzięki temu udało się wygenerować 35 typów planów mieszkań. Jednakże mimo znacznego potencjału schemat A. Sizy miał dwa poważne ograniczenia:

1. Utrudnione przekazywanie zasad kształtowania rzutów innym projektantom, ponieważ system nie był nigdy wcześniej stosowany w praktyce, a ponadto nie był dostatecznie sformalizowany.

2. Utrudnione prezentowanie zbioru rozwiązań spowodowane nieodpowiedniością tradycyjnych narzędzi projektowania i nowatorskiego podejścia projektowego. Wywoływało to poważne problemy w procesie komunikacji architekt - użytkownik.

W związku z powyższym generatywny potencjał systemu nie mógł być w pełni wykorzystany. J.P. Duarte dla przezwyciężenia powyższych problemów postanowił zastosować gramatykę form opartą na stylu A. Sizy. W gramatyce „Malagueira” forma matka to prostokąt z markerem „L”. Strategia kompozycji opiera się na pięciu zasadach sterowania prostokątami przedstawiającymi pomieszczenia mieszkania. Te reguły to: A - rozdzielanie, B - połączenie, G - rozszerzenie, H - przypisanie funkcji, I - zmiana funkcji. Funkcje pomieszczeń („fn”) oznaczone SA markerami. Markery służą również do identyfikacji ostatnio rozmieszczonej linii („v”) i pokazują, z której strony prostopadłościanu może zostać dokonane kolejne przyłączenie (z obu stron - reguła A lub tylko z jednej strony - B). Reguły A i B odnoszą się do przyłączeń prostopadłych do dłuższego boku prostokąta. Reguła C - do boku krótszego. Kolejne reguły określają możliwość przyłączania po przekątnej (D), blokowanie podziałów poprzez usunięcie markera (E), łączenie dwóch sąsiednich prostopadłościanów w celu utworzenia jednego większego (F), powiększenie pomieszczenia kosztem pomieszczenia sąsiedniego (G), funkcję pomieszczenia (H), zmianę funkcji dwóch sąsiadujących ze sobą pomieszczeń. Wszystkie te reguły definiują organizację funkcjonalną kondygnacji.

Program J.P. Duarte bazuje na zastosowaniu zasad gramatyki deskryptywnej (*description grammar*) i gramatyki form (*shape grammar*). Gramatyka form tworzy rzut obiektu, a reguły deskryptywne odnoszą się do opisu funkcji. Zgodnie z twierdzeniem Stiny’ego, który uważa, że gramatyka deskryptywna może być zastąpiona przez gramatykę innego języka i że możliwa jest translacja w obu kierunkach, J.P. Duarte wykorzystuje taki mechanizm przekształceń, który zapewnia możliwość otrzymania (ekstrakcji) projektu na podstawie jego opisu.

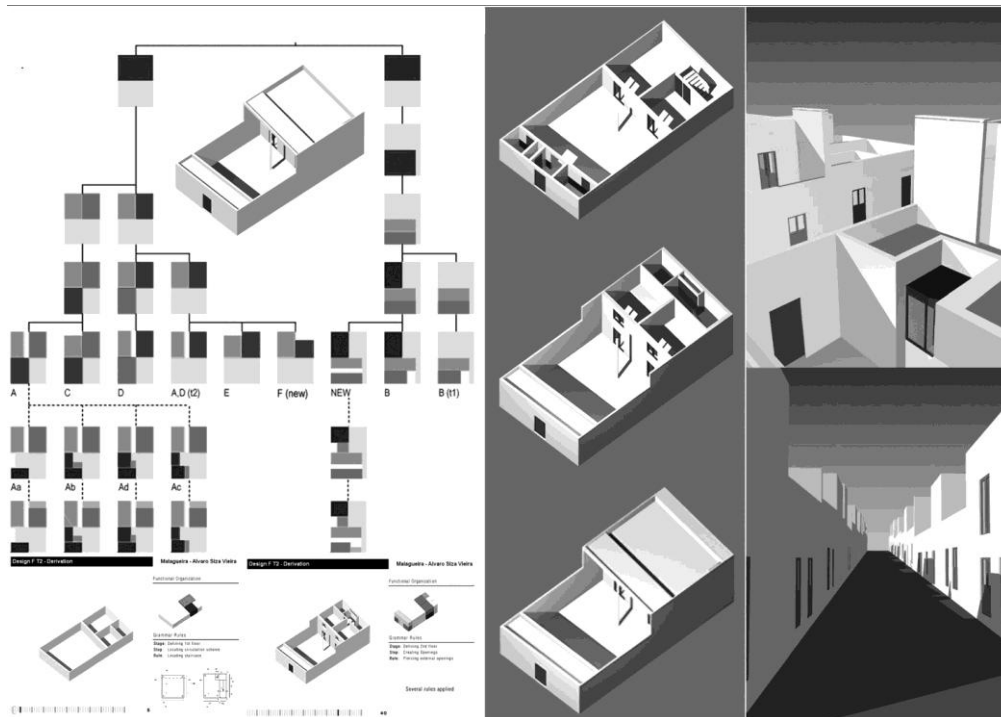
Program J.P. Duarte zawiera 11 parametrów projektowych:

1. Tabela minimalnych i maksymalnych wielkości stref funkcjonalnych.
2. Tabela minimalnych i maksymalnych wielkości pomieszczeń (zgodnie z obowiązującymi normami).
3. Wykaz pozostałych wielkości, takich jak wysokość kondygnacji, grubość stropu, ilość stopni w biegu schodów.
4. Opis lokalizacji (przestrzeni przylegających do obiektu z każdej strony).
5. Typ domu (z przedogródkiem lub z ogrodem za domem).
6. Liczba kondygnacji.
7. Rozmieszczenie balkonów.
8. Wykaz stref funkcjonalnych budynku (dziedziniec wewnętrzny, sypialnie, pokój dzienny i pomieszczenia pomocnicze).
9. Wykaz pomieszczeń.
10. Macierz określająca sąsiedztwo stref i pomieszczeń domu.
11. Hierarchia potrzeb użytkownika.

Na podstawie powyższych zasad został zaprojektowany interaktywny program do generowania i analizy rzutów domu. Składa się on z dwóch modułów. Pierwszy moduł - „Programator” (*Programmer*) - generuje program mieszkania na podstawie danych wejściowych określonych przez użytkownika. Drugi - „Interpretator” (*Interpreter*) - generuje rozwiązanie funkcjonalne zgodne z danymi modułu „Programator”. „Programator” tworzy również wykaz przestrzeni funkcjonalnych, oblicza ich powierzchnie, opisuje topologię oraz opracowuje kosztorys. Użytkownik może zmieniać program mieszkania, odejmując lub dodając pomieszczenia, zmieniając topologie oraz indywidualne wymagania dla każdej przestrzeni.

Zmiany te są możliwe tylko do momentu osiągnięcia minimalnych parametrów dla danego typu mieszkań. Zabezpiecza to przed powstawaniem rozwiązań substandardowych. Użytkownik może zmienić dowolny parametr i przeprowadzić ponowną kalkulację dla określenia poziomu komfortu mieszkania.

„Interpretator” zaczyna działać w ramach gramatyki „Malagueira”, gdy osiągnie ona maksymalny możliwy poziom zgodności z wymaganiami zawartymi w programie funkcjonalnym mieszkania.



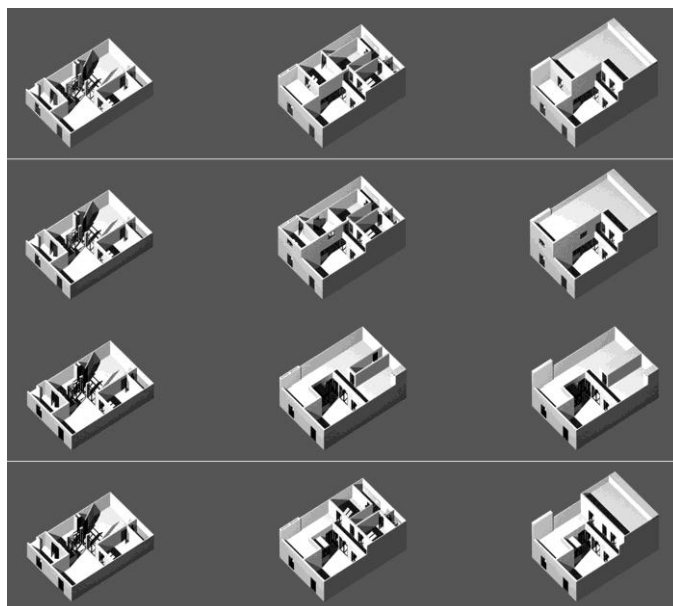
Ryc. 2. Przykłady rozwiązań wygenerowanych przez system Malagueira, autor J.P. Duarte

Użytkownik w dalszym ciągu może w każdym momencie zmienić dowolny parametr oraz wprowadzić zmiany w programie mieszkania. Po przeprowadzeniu zmian system generuje rozwiązanie dla nowego programu. Procedura może być powtarzana do chwili, gdy użytkownik otrzyma zestaw wariantów umożliwiających przeprowadzenie przez niego wyboru. Wybrane rozwiązanie jest przedstawiane jako przestrzeń wirtualna oraz jako makieta, wyprodukowana za pomocą trójwymiarowej drukarki (*rapid prototyping*). Pozwala to na szczegółowe zapoznanie się z projektem mieszkania.

Gramatyka form jest ściśle związana z twórczym aspektem projektowania, gdyż jak twierdzi J. Gero, twórcze projektowanie może być opisane jako takie pobudzenie systemu, dzięki któremu możliwe jest uzyskanie nieoczekiwanych i niemożliwych do przewidzenia rezultatów (Gero, Maher, 2001). Aspekt ten zajmuje centralne miejsce w gramatyce form. Przewidzenie rezultatu zastosowania reguł

gramatyki jest praktycznie niemożliwe, gdyż zdolność generatywna gramatyki form jest niezwykle wysoka (Knight, 1998). Projektant może eksperymentować z nowymi rozwiązaniami, wynikającymi z przyjętej przez niego przestrzennej organizacji. Proces projektowania staje się procesem adaptacyjnym, w rezultacie którego powstaje macierz rozwiązań będących inwariantem rozwiązania podstawowego.

Gramatyka form może być traktowana jako działania twórcze także i w aspekcie pragmatycznym. Po stworzeniu słownika form projektant musi świadomie podjąć decyzję o wyborze poszczególnych form. Może tego dokonać nie tylko na podstawie przesłanek wynikających z kontekstu funkcjonalnego, ale również na podstawie przesłanek estetycznych. Warunki lokalizacji i ograniczenia projektowe mogą wymusić na projektancie sprawdzenie kombinacji innych form. Wybór formy może odpowiadać stylowi prac danego projektanta lub też stylowi epoki.



Rys. 3. Warianty domu typu Ab wygenerowane przez system Malagueira, autor J.P. Duarte

PODSUMOWANIE

Przedstawione w referacie dwa programy wspomagające projektowanie architektury mieszkaniowej nie zostały, niestety, wdrożone do praktyki projektowej na szerszą skalę. Metoda Y. Friedmana pozostała metodą wyłącznie teoretyczną. Nie pozostała ona jednak bez wpływu na metodologię projektowania architektonicznego. Jej znaczenie polega na aktywnym włączeniu użytkownika w proces projektowania. Obecnie projektowanie współuczestniczące (*participatory designing*) jest powszechnie akceptowane i stosowane w praktyce projektowej. Program oparty na gramatyce form ma szansę na dalszy rozwój, gdyż bez wątpienia gramatyka form jest jednym z najbardziej efektywnych narzędzi do poszukiwania nowych form architektonicznych. Jest ona obszarem intensywnych badań naukowych w wielu ośrodkach. Szczegółowe informacje można znaleźć na stronie: (www.shapegrammar.org/people.html).

LITERATURA

1. Alexander C. (1963), *HIDECS-3: Four Computer Programs for the Hierarchical Decomposition of Systems which have an Associated Linear Graph*, Research Report, No R63-27, MIT Civil Engineering Systems Laboratory.
2. Alexander C. (1964), *Notes on the Synthesis of Form*, Harvard Univ. Press, Cambridge.
3. Duarte J.P. (2001) *Customizing mass housing: a discursive grammar for Siza's Malagueira houses: A Thesis for the PhD. in Architecture*, MIT, Dept. of Architecture.
4. Friedman Y. (1975) *Toward a scientific architecture*, The MIT Press, Cambridge.
5. Gero J.S., Maher M.L. (2001) *Computational and Cognitive Models of Creative Design V*, Key Centre of Design Computing and Cognition, Sydney: University Sydney.
6. Knight T.W. (1998) *Designing a Shape grammar: Problems of Predictability*, [w:] J.S. Gero, F. Sudweeks (eds.) *Artificial Intelligence in Design - Dordrecht: Kluwer Academic Publishers*, s. 499-516.
7. Stiny G., Gips J. (1972) *Shape Grammars and the Generative Specification of Painting and Sculpture*, [w:] C.V. Freiman (ed.), *Proceedings of IFIP Congress71*, Amsterdam, North-Holland, s. 1460-1465
8. Stiny G. (1977) *Ice-ray: a note on Chinese lattice designs*, *Environment and Planning B* 4, s. 89-98.
9. Stiny G., Mitchell W. J. (1978) *The Palladian grammar*, *Environment and Planning B* 5, s. 5-18.
10. Rittel H., (1972) *Son of Rittelthink*, The DMG 5th Anniversary Report, DMG Occasional Paper, №1.
11. <http://dspace.mit.edu/handle/1721.1/8189>

Podziękowania

Autor dziękuje profesorowi Jose Pinto Duarte za udostępnienie ilustracji zamieszczonych w niniejszej pracy.