Weryfikacja procesów biznesowych metodą dedukcyjną z wykorzystaniem logiki temporalnej

Radosław Klimek
AGH Akademia Górniczo-Hutnicza, Wydział EAIiE, Katedra Automatyki

Streszczenie: Praca dotyczy formalnej analizy i weryfikacji modeli biznesowych wyrażonych w notacji BPMN. Weryfikacja oparta jest na wnioskowaniu dedukcyjnym. Jako metodę wnioskowania dla modeli biznesowych zaproponowano metodę tablic semantycznych, która cechuje się apagogicznością oraz analitycznością. Została przedstawiona metoda translacji podstawowych wzorców projektowych BPMN do formuł logiki temporalnej, stanowiących logiczną specyfikację analizowanego modelu. Zarówno logiczna specyfikacja, jak i właściwości badanych procesów są wyrażone formułami tzw. najmniejszej logiki temporalnej. Formuły te są następnie przetwarzane z wykorzystaniem metody tablic semantycznych. Innowacyjność proponowanego podejścia może istotnie wpłynąć na redukcję kosztów wytwarzania oprogramowania, ze względu na możliwość wykrycia błędów oprogramowania już w fazie jego modelowania, wyprzedzając tym samym znacznie fazy implementacji i testowania.

Słowa kluczowe: modele biznesowe, BPMN, SOA, wzorce projektowe, logika temporalna, wnioskowanie dedukcyjne, metoda tablic semantycznych, generowanie specyfikacji

Logika formalna zawsze zajmowała istotne miejsce w życiu człowieka, stanowiąc naturalne środowisko towarzyszące w jego życiu codziennym przy podejmowaniu różnych decyzji. Badając związki pomiędzy prawdziwością a fałszywością zdań ze względu na ich strukturę, umożliwia zarówno formalne, jak i sprawne podejście do procesu wnioskowania. Logika temporalna, będąca ważnym fragmentem logiki klasycznej, odgrywa kluczową rolę w analizie i weryfikacji systemów informatycznych, przy czym istnieją tutaj dwa zasadniczo odmienne podejścia [7]. Pierwsze bazuje na eksploracji przestrzeni stanów i jest to tzw. weryfikacja modelowa (ang. *model checking*) [6], a drugie na klasycznym wnioskowaniu dedukcyjnym z wykorzystaniem logiki formalnej. Aczkolwiek oba podejścia są dobrze rozpoznane, to jednak tylko w przypadku pierwszego z nich dokonał się w ostatnich latach znaczący, a może nawet spektakularny postęp w zakresie opracowania odpowiednich metod i algorytmów, podczas gdy drugie z nich wciąż nie ma liczących się i udokumentowanych wyników. Tymczasem warto zauważyć, że co prawda nie znając zasad logiki, także można przeprowadzać rozumowanie, podobnie jak – na zasadzie analogii – nie znając zasad gramatycznych, można się posługiwać językiem, to nie ulega jednak wątpliwości, że znajomość reguł gramatycznych podnosi kulturę i jakość myślenia i wypowiedzi w danym języku, tak znajomość i posługiwanie się regułami logiki zwiększa wiarygodność procedury dedukcyjnej.

Modelowanie biznesowe odgrywa coraz większe znaczenie, umożliwiając zrozumienie zarówno bieżącej aktywności, ale także potencjału ewentualnych zmian i ulepszeń w różnych planach i działaniach. Modelowanie biznesowe może także istotnie i korzystnie wpłynąć na zsynchronizowanie bieżącej aktywności z modelem informatycznym. Integracja taka jest niezbędna, o ile system informatyczny ma wspierać system biznesowy. Jako narzędzie do modelowania procesów biznesowych najbardziej ugruntowaną pozycję ma notacja BPMN.

Celem pracy jest przedstawienie kompletnej architektury systemu wnioskowania, umożliwiającego formalną weryfikację pewnej klasy systemów informatycznych. Weryfikacji poddawane są procesy biznesowe wyrażone w notacji BPMN – modele te stanowią swego rodzaju logiczne sieci działań i wydaje się, że są one szczególnie dogodne w procesie pozyskiwania specyfikacji logicznej analizowanego systemu. Zarówno do specyfikacji systemu, jak i wyrażenia badanych i pożądanych własności, wykorzystywana jest logika temporalna. Pokazano możliwości automatyzacji zarówno procesu pozyskiwania specyfikacji, jak i samego dowodzenia właściwości systemu.

1. Procesy biznesowe i wzorce projektowe

Procesy biznesowe mają w informatyce coraz większe znaczenie. *Procesy biznesowe* możemy rozumieć jako uporządkowane i powiązane ze sobą aktywności i zadania pozwalające na zrozumienie planów działań i przedsięwzięć. Celem procesu biznesowego jest rozwiązanie określonego problemu lub osiągnięcie pewnego zamierzonego efektu. Najbardziej popularną notacją dla modeli biznesowych jest notacja BPMN (ang. *Business Process Modeling Notation*), która została zaprojektowana przez Business Process Management Initiative (BPMI), jako standard notacyjny w modelowaniu takich procesów. Należy zwrócić uwagę, że BPMN w kontekście architektur usług sieciowych i paradygmatu SOA może stanowić ważną warstwę wizualizacyjną i w ten sposób zmniejszyć dystans pomiędzy procesem projektowania a środowiskiem wykonawczym SOA, co wynika także z możliwości odpowiedniej translacji (por. np. [12, 16]). Szczegółowe informacje na temat notacji BPMN można znaleźć w wielu pracach BPMN-2009.

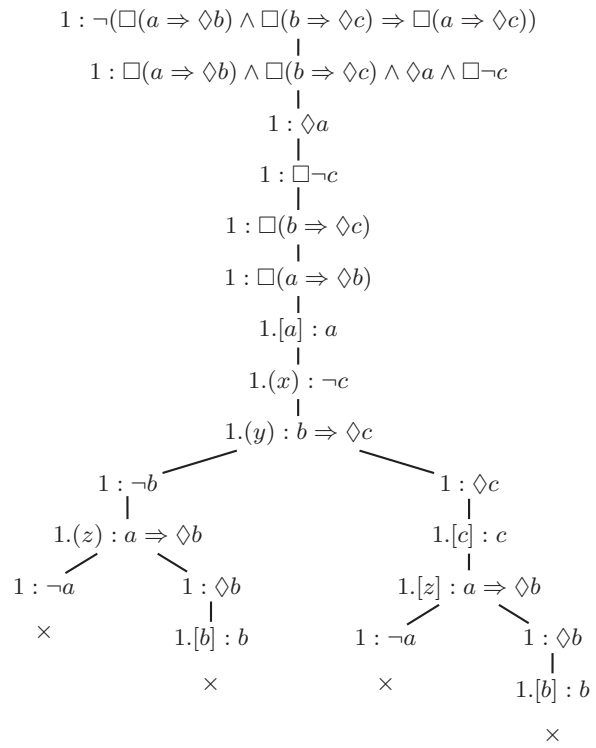
Z modelowaniem biznesowym wiąże się często koncepcję tzw. wzorców projektowych, które odgrywają istotną rolę w samym modelowaniu procesów biznesowych. *Wzo-*

rzec jest „abstrakcją postaci konkretnej formy, która pojawia się w dowolnym momencie" [14]. Wzorzec projektowy jest ogólnym opisem pewnej struktury aktywności, użycie której rozwiązuje pewien probelm w obszarze modelowania biznesowego. Wszystkie wzorce zostały skatalogowane i zawierają łącznie 23 obiekty [1, 15]. Stosowanie wzorców projektowych wpływa korzystnie na strukturę projektowanego modelu biznesowego, a z punktu widzenia podejścia przedstawionego w tej pracy umożliwia także automatyzację procesu budowy specyfikacji logicznej analizowanego modelu.

2. Logiczne preliminaria

Najważniejszymi elementami składającymi się na stosowany aparat formalny jest logika temporalna oraz wnioskowanie metodą tablic semantycznych. *Logika temporalna* jest niekwestionowanym i wygodnym formalizmem umożliwiającym specyfikację i weryfikację systemów współbieżnych i reaktywnych (por. np. [8, 10]). Formuły logiki temporalnej skutecznie wyrażają własności żywotności i bezpieczeństwa systemów, mogą odgrywać kluczową rolę w ich dowodzeniu. Logika temporalna istnieje w wielu odmianach i karnacjach, aczkolwiek w dalszych rozważaniach skupimy się na aksjomatycznym i dedukcyjnym systemie dla tzw. *najmniejszej logiki temporalnej* (por. np. [3]). Logika ta jest znana także jako logika temporalna klasy K, i może być rozszerzona poprzez wprowadzenie szeregu nowych własności struktury czasowej [5, 13]. Przykładami takich wzbogaconych logik są: logika/formuła D: (przykładowa formuła) $\Box p \Rightarrow \Diamond p$; logika/formuła T: $\Box p \Rightarrow p$; logika/formuła G: $\Diamond\Box p \Rightarrow \Box\Diamond p$; logika/formuła 4: $\Box p \Rightarrow \Box\Box p$; logika/formuła 5: $\Diamond p \Rightarrow \Box\Diamond p$; logika/formuła B: $p \Rightarrow \Box\Diamond p$; etc. Należy zauważyć, że możliwe jest także połączenie poszczególnych własności oraz logik, i w ten sposób ustanowienie relacji pomiędzy nimi, np.: KB4 $\Leftrightarrow$ KB5, KDB4 $\Leftrightarrow$ KTB4 $\Leftrightarrow$ KT45 $\Leftrightarrow$ KT5 $\Leftrightarrow$ KTB.

Jak już wspomniano, przyjęta tutaj metoda wnioskowania jest w pewnej opozycji zarówno do metody weryfikacji modelowej, związanej z eksploracją stanów, jak i metody dedukcyjnej opartej o rezolucję. Metoda ta, zwana *metodą tablic semantycznych* jest znana z logiki klasycznej, ale może być także stosowana na gruncie logik modalnych [2]. Metoda cechuje się apagogicznością i analitycznością. W każdym kroku dobrze zdefiniowanej procedury postępowania otrzymujemy formuły coraz prostsze i składające się z coraz mniejszej liczby elementów, z których usuwane są spójniki logiczne. Na końcu tak przeprowadzonej procedury dekompozycyjnej przeszukiane są wszystkie gałęzie otrzymanego drzewa celem odnalezienia sprzeczności w poszczególnych jego gałęziach. Jeżeli w gałęzi drzewa znajdują się sprzeczności, to gałąź taką uznaje się za *domkniętą*. Jeżeli wszystkie gałęzie są domknięte, to wynika z tego, że prawdziwa jest formuła początkowa. Dużą wartość informacyjną, oczywiście w przypadku niepowodzenia w dowodzeniu własności, stanowią tzw. otwarte gałęzie drzewa wnioskowania, gdyż poprzez nie otrzymujemy zbiór zdań elementarnych, które nie spełniają własności systemu. Na rys. 1 zostało pokazane drzewo wnioskowania dla przykładowej formuły $\Box(a \Rightarrow \Diamond b) \wedge \Box(b \Rightarrow \Diamond c) \Rightarrow \Box(a \Rightarrow \Diamond c)$. Każda formuła jest

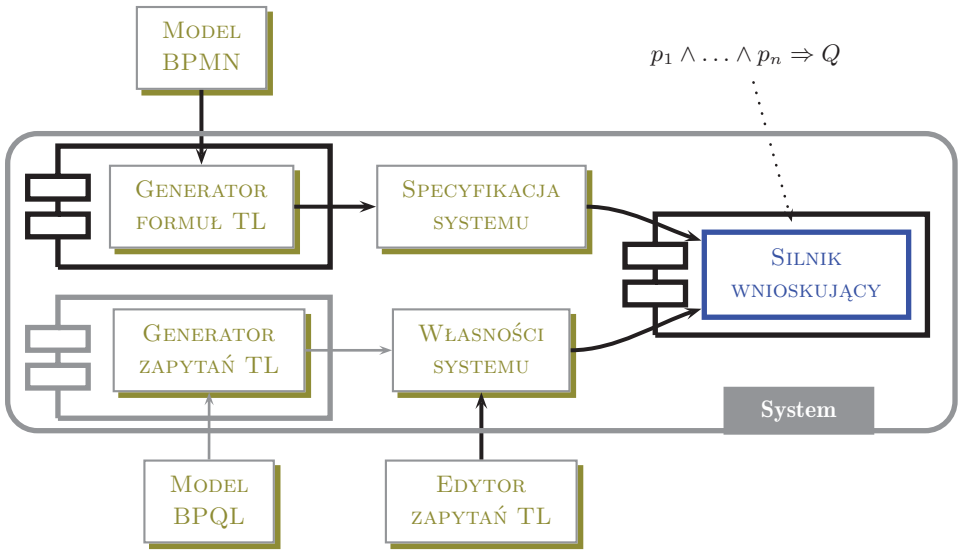


Rys. 1. Drzewo wnioskowania metodą tablic semantycznych
Fig. 1. Inference tree using semantic tableaux method

poprzedzona etykietą odnoszącą się do aktualnie wyróżnionego świata. Świat początkowy, w którym jest prawdziwa dana formuła, reprezentowany jest przez (fragment) etykiety „1 :". Etykieta „1.(x)", gdzie x jest zmienną wolną, reprezentuje wszystkie możliwe światy będące następnikami świata 1. Z kolei napis „1.[p]", gdzie p jest formułą atomową, reprezentuje jeden z możliwych światów, następników świata 1, gdzie formuła p jest prawdziwa. Przyjęty tutaj sposób postępowania, a także etykietowania, nawiązuje do rachunku kwantyfikatorów pierwszego rzędu i można go znaleźć np. w pracy [9]. Wszystkie gałęzie przykładowego drzewa są domknięte (×), gdyż we wszystkich znajdują się sprzeczności. Generalnie można wykazać, że metoda tablic semantycznych jest poprawna, a algorytm budowy drzewa zawsze zatrzymuje się.

3. System dedukcyjny

Proponowany system wnioskowania z wykorzystaniem metody tablic semantycznych dla modeli biznesowych BPMN został przedstawiony na rys. 2, podobny także [11]. System składa się z kilku komponentów. Pierwszy zapewnia funkcjonalność generującą specyfikację logiczną, rozumianą jako zbiór formuł logiki temporalnej (klasy K). Formuły są generowane bezpośrednio ze wzorców projektowych BPMN i gromadzone w systemie w osobnym module. Powiązane ich później symbolem koniunkcji daje ogólną specyfikację systemu: $p_1 \wedge \ldots \wedge p_n = P$, wchodzącą w skład wejścia procesu wnioskującego. Kolejny komponent przechowuje pożądane i badane własności systemu Q, najprościej wprowadzone poprzez dowolny edytor (tekstowy). (Innym sposobem pozyskiwania formuł służących do weryfikowania własności systemu jest język zapytań BPQL – ten komponent nie będzie tutaj jednak rozważany, a celem jego tu-



Rys. 2. Architektura systemu automatycznej weryfikacji modeli biznesowych metodą dedukcyjną
Fig. 2. Architecture of an automatic and deduction-based verification system of business models

taj przedstawienia jest zasygnalizowanie dalszego kierunku możliwych prac). Zarówno specyfikacja systemu, jak i badane własności stanowią wejście dla zasadniczego modułu, którym jest silnik wnioskujący. Silnik ten realizuje proces dedukcyjny metodą tablic semantycznych, a wejście dla niego stanowi formuła $P \Rightarrow Q$, albo ściślej:

$$p_1 \wedge \ldots \wedge p_n \Rightarrow Q \tag{1}$$

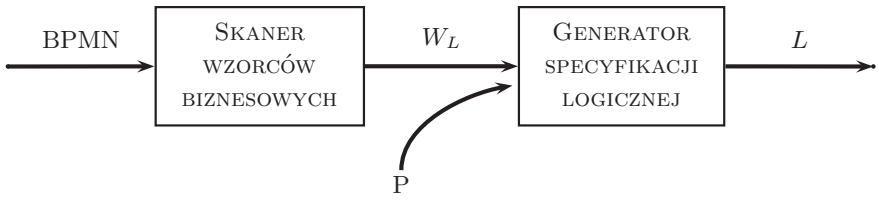
Po zanegowaniu formuły 1, umieszczana jest ona w korzeniu drzewa wnioskowania, a następnie poddawana jest procesowi dekompozycji zgodnie z regułami metody tablic semantycznych. Znalezienie sprzeczności we wszystkich gałęziach drzewa oznacza brak wartościowania spełniającego umieszczoną w korzeniu formułę zanegowaną. To oznacza, że drzewo jest domknięte, co prowadzi do stwierdzenia, że formuła początkowa 1 jest prawdziwa. Wszystkie komponenty systemu wnioskującego mogą być wykonywane wielokrotnie, uwaga ta dotyczy zarówno modułu pozyskiwania specyfikacji logicznej, np. po jej modyfikacji, także wprowadzania nowych, badanych formuł własnościowych, jak i samej pracy silnika wnioskującego.

4. Generowanie specyfikacji

Zbudowanie specyfikacji logicznej systemu, poprzez pozyskanie zbioru formuł liniowej logiki temporalnej opisujących jego własności, ma kluczowe znaczenie i stanowi swego rodzaju wąskie gardło w całym procesie dowodzenia własności systemów. Proponowana metoda ekstrakcji specyfikacji logicznej systemu bazuje na dobrze znanych wzorcach projektowtch BPMN, zakłada sie przy tym, że cały model jest zbudowany tylko i wyłącznie z takich wzorców. Budowa specyfikacji polega na wydzieleniu wzorców projektowych w modelu biznesowym, translacji wzorców do wyrażenia logicznego W_L, a następnie wygenerowaniu specyfikacji logicznej L. Schemat organizacyjny dla generatora formuł logiki temporalnej został przedstawiony na rys. 3, poniżej zostaną krótko omówione najważniejsze pojęcia.

Z każdym wzorcem projektowym związany jest predefiniowany zbiór formuł logiki temporalnej $pat(a_1, ..., a_n) = \{f_1, ..., f_m\}$ opisujący własności danego wzorca, przy czym a_i stanowią aktywności biznesowe, względnie inne zagnieżdżone wzorce, występujące w danym wzorcu, a f_i są formułami LTL nad aktywnościami elementarnymi. Przykładowo dla wzorca równoległego rozszczepienia (ang. *Paralell Split*) zbiór formuł ma postać: $ParSplt(a, b, c) = \{a \Rightarrow \Diamond b \wedge \Diamond c, \Box\neg(a \wedge (b \vee c))\}$. Predefiniowanie formuł logiki temporalnej dla każdego wzorca biznesowego umożliwia utworzenie zbioru predefiniowanego P z takimi właśnie formułami. Po przeanalizowaniu całego modelu biznesowego i wyodrębnieniu wzorców wchodzących w jego skład, zapisywany jest on w postaci *wyrażenia logicznego* W_L, który poprzez operacje sekwencji i zagnieżdżenia zawiera w sobie wszystkie zastosowane w modelu wzorce projektowe. Przykładowo, dla sekwencji złożenia dwóch wzorców wyłącznego wyboru (ang. *Exclusive Choice*) oraz prostego złączenia (ang. *Simple Merge*), otrzymamy wyrażenie logiczne $Seq(ExclCh(a, b, c), SimpMer(d, e, f))$.

Ostatnim krokiem postępowania jest zbudowanie *specyfikacji logicznej* L z wyrażenia logicznego W_L, przy czym $L(W_L) = \{f_i : i > 0\}$, gdzie f_i jest dowolną formułą logiki temporalnej, uzyskaną w wyniku procesu przekształcenia wyrażenia logicznego do specyfikacji logicznej. Algorytm generowania specyfikacji logicznej ma dwie dane wejściowe: wyrażenie logiczne W_L oraz predefiniowany zbiór P, będący zbiorem formuł logiki temporalnej dla każdego wzorca projektowego. Algortym przekształcenia W_L do L, przy założeniu wartości początkowej $L = \emptyset$ jako zbioru pustego, polega na wyszukaniu w pierwszej kolejności wzorców najbardziej zagnieżdżonych, i o ile argumentami takiego wzorca są aktywności elementarne, to następuje przepisanie formuł związanych z danym wzorcem do specyfikacji, tj. $L = L \cup pat()$. Jeżeli jakikolwiek wzorzec wyrażenia logicznego zawiera jako argumenty inne wzorce, to wówczas w miejsce argumentu jest podstawiana alternatywa logiczna wszystkich aktywności wewnętrznych. Przy-



Rys. 3. Generator formuł TL
Fig. 3. TL formulas generator

kładowo $ParSplt(Seq(a,b),c,d)$ prowadzi do specyfikacji $L = \{a \Rightarrow \Diamond b\} \cup \{(a \vee b) \Rightarrow \Diamond c \wedge \Diamond d, \Box\neg((a \vee b) \wedge (c \vee d))\}$.

5. Podsumowanie

Zostały przedstawione zagadnienia dotyczące formalnej weryfikacji modeli biznesowych, zapisanych w notacji BPMN, z wykorzystaniem podejścia dedukcyjnego i logiki temporalnej metodą tablic semantycznych, wraz z automatyzacją takiego podejścia. Należy także wspomnieć o zaawansowanych pracach implementacyjnych nad prototypowymi wersjami poszczególnych komponentów, składających się na proponowany system. Prace te są bliskie ukończenia.

Bibliografia

1. van der Aalst W.M.P., ter Hofstede A.H.M., Kiepusewski B., Barros A.P. (2003): *Workflow Patterns*. Distributed and Parallel Databases, 4(1), 2003, 5–51.
2. D'Agostino M., Gabbay D.M., Hähnle R., Posegga J. (eds) (1999): *Handbook of Tableau Methods*, Kluwer Academic Publishers 1999.
3. van Benthem J. (1995): *Temporal Logic*. Clarendon Press 1993–95, 241–350.
4. *Business Process Modeling Notation Specification*. Version 1.2, January 2009, OMG Document dtc/2009-01-03.
5. Chellas B. (1980): *Modal Logic*, Cambridge University Press 1980.
6. Clarke E.M.Jr., Grumberg O., Peled D.A. (1999): *Model Checking*. MIT Press 1999.
7. Clarke E.M., Wing J.M. et al. (1996): *Formal methods: State of the art and future directions*, ACM Computing Surveys, 28(4), 1996, 626–643.
8. Emerson E.A. (1990): *Temporal and Modal Logic*, Elsevier, MIT Press 1990, 995–1072.
9. Hähnle R. (1998): *Tableau-based Theorem Proving*. ESSLLI Course 1998.
10. Klimek R. (1999): *Wprowadzenie do logiki temporalnej*, Wydawnictwa Naukowo-Techniczne AGH 1999.
11. Klimek R., Skrzyński P., Turek M. (2010): *Deduction based verification of business models*, [w:] Korczak J., Dudycz H., Dyczkowski M. (red.): Advanced Information Technologies for Management, (Research Papers of Wroclaw University of Economics, 147), Publishing House of Wrocław University of Economics 2010, 173–188.
12. Ouyang C., Dumas M., ter Hofstede A.H.M., van der Aalst W.M.P. (2006): *From BPMN Process Models to BPEL Web Services*. IEEE International Conference on Web Services (ICWS'06), IEEE Computer Society 2006, 285–292.
13. Pelletier F.J. (1993): *Semantic Tableau Methods for Modal Logics that Include the B and G Axioms*. AAAI Technical Report FS-93-01, 1993.
14. Riehle D., Zullighoven H. (1996): *Understanding and Using Patterns in Software Development*. Theory and Practice of Object Systems, 2(1), 1996, 3–13.
15. Russell N., ter Hofstede A.H.M., van der Aalst W.M.P., Mulyar N. (2006): *Workflow Control-Flow Patterns: A Revised View*, BPM Center Report BPM-06-22, BPMcenter.org, 2006.
16. White S.A. (2005): *Using BPMN to Model BPEL Process*. BPTrends, 2005, 3, 1–18.

Deduction-based Formal Verification of Business Models using Temporal Logic

Abstract: The paper concerns formal analysis and verification of business models expressed in BPMN. This verification is based on a deductive reasoning. As a method of inference for business models semantic tableaux method is proposed. Automatic transformations of the basic BPMN workflow patterns to temporal logic formulas are proposed. These formulas constitute a logical specification of the analyzed model. Both the logical specification and the desired system properties are expressed as formulas of the smallest linear temporal logic. These formulas are later processed using semantic tableaux method. Applying this innovative concept might result in software development costs reduction as some errors might be addressed in the modeling phase not in implementation or testing phase.

Keywords: business models, BPMN, SOA, workflow design patterns, temporal logic, deductive reasoning, semantic tableaux method, generating specifications

dr inż. Radosław Klimek

Zainteresowania naukowe: metody formalne, logika temporalna, klasyczne i nieklasyczne metody wnioskowania, inżynieria oprogramowania, algorytmika, teoria języków programowania. Prywatnie: muzyka klasyczna (i nie tylko), malarstwo, fotografia, ogród japoński, ogród barokowy.
e-mail: rklimek@agh.edu.pl