

dr hab. inż. Jerzy Zając, Prof. PK
mgr inż. Grzegorz Chwajół
Politechnika Krakowska

MODELOWANIE WYBRANYCH ZAGADNIĘŃ W ROZPROSZONYM SYSTEMIE STEROWANIA PRODUKCJĄ

Modelowanie jest jednym z istotnych etapów procesu projektowania i sterowania współczesnych systemów produkcyjnych. W niniejszej pracy dokonano krótkiej charakterystyki stosowanych metod modelowania oraz przedstawiono wykonane przy użyciu języka UML modele wybranych zagadnień z zakresu rozproszonych systemów sterowania produkcją.

MODELING OF SELECTED ASPECTS OF DISTRIBUTED PRODUCTION CONTROL SYSTEM

Modeling plays an essential role in the process of production systems design and control. The paper briefly describes methods used for manufacturing systems modeling and presents UML models of selected aspects of distributed production control systems.

1. WPROWADZENIE

Złożoność współczesnych systemów produkcyjnych sprawia, iż na etapie ich projektowania niezbędne staje się opracowanie modeli przedstawiających różne aspekty ich budowy i funkcjonowania. Modele takie, będące uproszczoną – umyślnie i celowo – reprezentacją rzeczywistości [5], pozwalają m.in. na usystematyzowanie wiedzy na temat projektowanego systemu, ułatwiają definiowanie i analizę wzajemnych powiązań pomiędzy poszczególnymi jego elementami oraz stanowią ważne źródło poznania tego systemu.

Problematyka modelowania dyskretnych systemów produkcyjnych realizujących procesy współbieżne posiada bardzo bogatą bibliografię. Podstawą większości metod modelowania takich systemów jest badanie zmian stanu systemu. Model służący do symulacji działania systemu produkcyjnego powinien umożliwiać obserwowanie sekwencji zdarzeń związanych z rozpoczynaniem i kończeniem czynności elementarnych przebiegających względem czasu zgodnie z porządkiem mogącym zaistnieć w rzeczywistości. Model służący do sterowania przebiegiem realizowanego procesu produkcyjnego, aby być użytecznym, musi mieć zdolność do przetwarzania danych w czasie rzeczywistym, współbieżnie z procesem sterowanym, i dawać, na każdym etapie działania systemu produkcyjnego, podstawy do podejmowania decyzji dotyczących dalszego sterowania poprzez umożliwienie wyboru czynności, którą należy zlecić do wykonania.

Najczęściej wykorzystywanym sposobem modelowania procesów dyskretnych są Sieci Petriego [4], jednak zastosowanie ich w systemach rozproszonych napotyka trudności ze względu na brak oddzielnej reprezentacji poszczególnych obiektów tworzących skład systemu. W klasycznej sieci Petriego jedno miejsce może bowiem reprezentować równocześnie stan kilku obiektów, co utrudnia bieżące śledzenie stanów pojedynczych obiektów systemu. Aby ominąć te trudności, na Politechnice Krakowskiej opracowano zmodyfikowaną wersję sieci, zwaną obiektowo-obszerną siecią Petriego [3].

Inny sposób modelowania, nazwany Modelem Macierzowym, przedstawiono w pracy [2]. Model ten umożliwia prosty i przejrzysty zapis dowolnej struktury systemu oraz reguł jego

działania. Model Macierzowy może być wykorzystany zarówno do symulacji jak i sterowania o charakterze scentralizowanym.

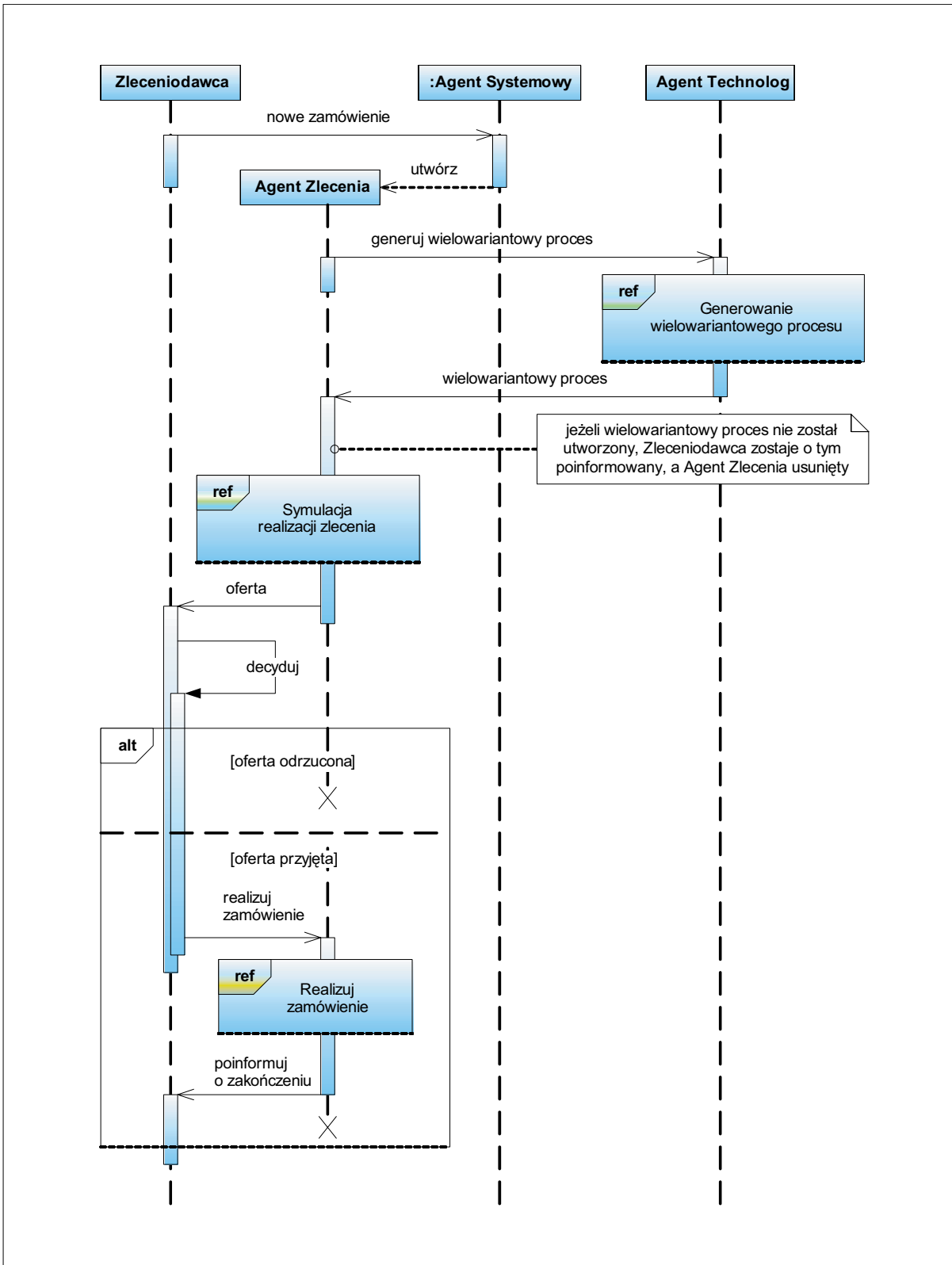
W ciągu ostatniego dziesięciolecia rozwinęły się w sposób znaczący narzędzia umożliwiające modelowanie procesu opracowania złożonych systemów oprogramowania. Jednym z efektów tych prac jest powstanie ujednoczonego języka modelowania UML (ang. *Unified Modeling Language*) [1]. UML jest graficznym językiem do wizualizacji, specyfikacji, projektowania i dokumentowania systemów oprogramowania. Może on być wykorzystany zarówno do opisu ogólnych relacji o charakterze koncepcyjnym, jak i szczegółowych zależności wykorzystywanych przez konkretny język programowania. Język ten jest coraz częściej wykorzystywany również w procesie modelowania systemów produkcyjnych. Powstanie UML było następstwem wieloletnich poszukiwań uniwersalnego narzędzia modelowania złożonych systemów oprogramowania, opracowywanych za pomocą obiektowo-zorientowanych języków programowania. Zdaniem twórców języka UML zrozumienie działania złożonego systemu wymaga przeanalizowania kilku dopełniających się, powiązanych ze sobą perspektyw (punktów widzenia). Każda z tych perspektyw może opisywać zarówno strukturę, jak i zachowanie systemu. W języku UML do przedstawienia punktów widzenia na działanie złożonego systemu wykorzystuje się diagramy. W ostatnim czasie zaobserwować można rosnące zainteresowanie wykorzystaniem UML w procesach modelowania systemów sterowania wytwarzaniem [6][7].

W niniejszej pracy zaprezentowano trzy modele przedstawiające wybrane aspekty z szerokiego zakresu zagadnień związanych z rozproszonymi systemami sterowania produkcją. Jako narzędzie modelowania wykorzystano język UML. Modele te wykorzystywane są do opracowania oprogramowania wieloagentowego systemu sterowania wytwarzaniem AIM (ang. *Agent Integrated Manufacturing*)[9]. W systemie AIM wyróżnić można kilka zasadniczych typów agentów stanowiących podstawę działania tego systemu. Są to m.in.: *agenty wykonawcze* reprezentujące poszczególne zasoby wytwórcze oraz związane z nimi *agenty dostosowujące* pełniące rolę pośredników w komunikacji ze sterownikami urządzeń; *agenty przedmiotowe*, których funkcją jest reprezentowanie wytwarzanych w systemie przedmiotów; *agent technolog*, którego zadaniem jest opracowywanie procesów technologicznych; *agenty systemowe* administrujące i monitorujące stan systemu. Na szczególną uwagę w aspekcie obsługi zamówień zasługują ostatnie wchodzące w skład systemu AIM agenty, tzw. *agenty zleceń*. Ich funkcją jest reprezentowanie poszczególnych typów produktów realizowanych w systemie wytwarzania. *Agent zlecenia* przechowuje informacje dotyczące zamówienia, takie jak: liczba sztuk produktu, termin i koszt jego wykonania, dokumentacja konstrukcyjna wraz z modelem CAD itp. Jego rolą jest także gromadzenie danych informujących o aktualnym stanie realizowanego zamówienia. Pierwszym aspektem dotyczącym rozproszonego systemu sterowania produkcją i omówionym w niniejszej pracy jest cykl życia agenta zlecenia.

2. MECHANIZM OBSŁUGI ZLECENIA

Agent zlecenia zostaje utworzony za pośrednictwem *agenta systemowego* z chwilą wprowadzenia do systemu danych określonych przez zleceniodawcę a dotyczących wymagań w stosunku do zamawianego produktu. Dane te wprowadzić można np. poprzez odpowiedni formularz na stronie www.

Fizyczna lokalizacja agenta zlecenia określana jest w zależności od przyjętego algorytmu, który obok inicjalizacji agenta na dedykowanym odgórnie komputerze dopuszcza także możliwość wykorzystania w tym celu komputera charakteryzującego się w danej chwili najmniejszym współczynnikiem obciążenia procesora.



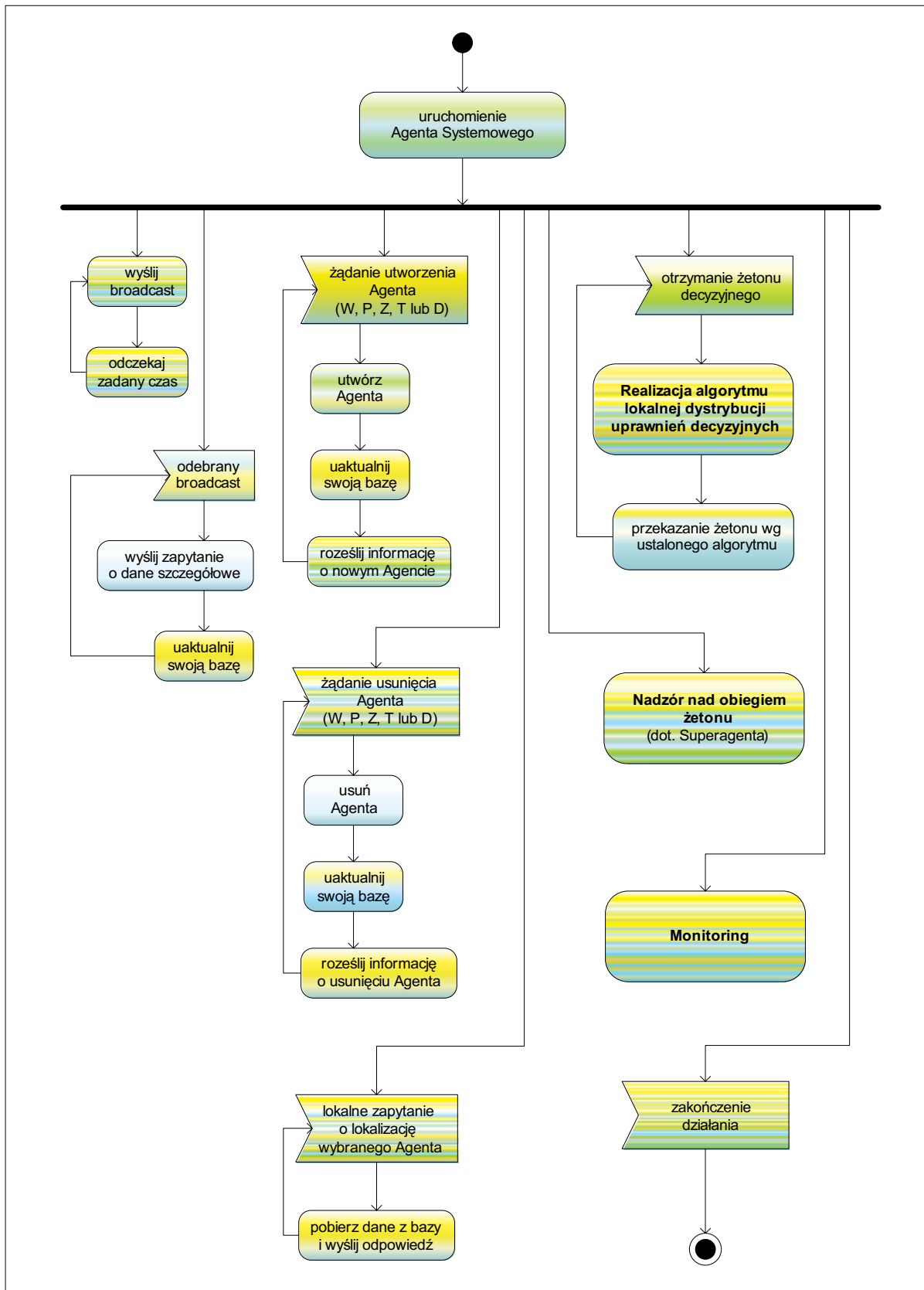
Rys. 1. Cykl życia agenta zlecenia

Pierwszym zadaniem nowoutworzonego *agenta zlecenia* jest przekazanie posiadanych danych konstrukcyjnych zamawianego produktu *agentowi technologowi*, który następnie – bazując na wiedzy dotyczącej dostępnych zasobów wytwórczych i ich możliwości technologicznych – przeprowadza działania mające na celu przygotowanie wielowariantowego procesu produkcyjnego. W efekcie, w typowej sytuacji *agent zlecenia* otrzymuje opis wielowariantowego procesu wytwórczego stanowiący podstawę do dalszych działań zgodnie z opracowanym algorytmem. Należy jednak na tym etapie uwzględnić także drugą sytuację, jaką jest brak możliwości wygenerowania procesu produkcyjnego, wynikający np. z awarii urządzeń wytwórczych czy też niedostępności niezbędnych narzędzi. W takiej sytuacji zleceniodawca jest informowany o zaistniałym fakcie, natomiast *agent zlecenia* zostaje usunięty z systemu. Założywszy realizowalność procesu produkcyjnego, w kolejnym etapie działań *agent zlecenia* inicjuje proces symulacji realizacji zamówienia. Przeprowadzana symulacja uwzględnia aktualnie realizowane w systemie procesy wytwórcze. W efekcie, etap ten ma na celu dostarczenie danych wymaganych do sformułowania oferty dla zleceniodawcy. Najistotniejszymi parametrami w tym przypadku są: szacowany termin oraz koszt realizacji zlecenia, przy czym, uwzględniając wielowariantowość realizacji procesu wytwórczego, a także pożądaną przez zleceniodawcę termin realizacji zamówienia, wskazane jest by w wyniku symulacji uzyskiwać więcej niż jedną parę wartości: termin i odpowiadający mu koszt. Przygotowana oferta jest następnie przedstawiana zlecającemu wykonanie produktu klientowi. Dalszy przebieg działań zależy od decyzji zleceniodawcy. W przypadku braku akceptacji warunków przedstawionych w ofercie, proces obsługi zamówienia zostaje zakończony, jednocześnie związany z nim *agent zlecenia* jest usuwany z systemu. Jeżeli jednak klient aprobuje określony w ofercie wariant realizacji zlecenia, zamówienie zostaje przekazane do realizacji. W tej fazie rola agenta reprezentującego zlecenie sprowadza się głównie do monitorowania stanu realizacji zamówienia oraz przekazywania stosownych informacji na żądanie operatora. Po zakończeniu realizacji procesu produkcyjnego zleceniodawca zostaje o tym fakcie poinformowany, zaś sam proces obsługi zlecenia zostaje zakończony, jednocześnie reprezentujący zlecenie agent jest usuwany z systemu. Rys. 1 przedstawia UML-owy diagram sekwencji obrazujący pełny cykl życia *agenta zlecenia*.

Drugim aspektem dotyczącym rozproszonego systemu sterowania produkcją i omówionym w niniejszej pracy są zadania *agenta systemowego*.

3. ZADANIA AGENTÓW SYSTEMOWYCH

Agenty systemowe w systemie AIM są odpowiedzialne za działania o charakterze administracyjno-monitorującym. Przyjęto, że ich liczba równa jest liczbie komputerów, które tworzą system sterowania, przy czym na jednym komputerze znajduje się jeden *agent systemowy*. W zakresie działań omawianych agentów wyróżnić można cztery zasadnicze grupy zadań. Zadania z pierwszej grupy wiążą się z działaniami administracyjno-informacyjnymi dotyczącymi pracy agentów wszystkich pozostałych typów uruchomionych na tym samym komputerze co *agent systemowy*. W zakres działań *agentów systemowych* wchodzi tu takie zadania jak: inicjalizacja poszczególnych agentów (wykonawczego, przedmiotowego, zlecenia, dostosowującego oraz technologa) i odpowiednio usuwanie z systemu tych, które na danym etapie procesu kończą realizację swych zadań, a także rejestracja poszczególnych agentów w lokalnej bazie adresowej, dystrybucja informacji lokalizacyjnych pomiędzy *agentami systemowymi* oraz udzielanie informacji w odpowiedzi na stosowne zapytania ze strony lokalnych agentów.



Rys. 2. Zadania agenta systemowego

Druga grupa zadań związana jest z udziałem *agentów systemowych* w procesie dystrybuowania uprawnień decyzyjnych. Zagadnienie to ma istotne znaczenie dla poprawności realizowanych w systemie AIM procesów decyzyjnych i będzie szerzej omówione w kolejnym rozdziale niniejszej pracy. Trzecią grupę zadań stanowią mechanizmy odpowiedzialne za badanie otoczenia sieciowego pod kątem poszukiwań innych *agentów systemowych*, a tym samym innych komputerów, na których mogą być zlokalizowane agenty wymagane w czasie realizacji procesu sterowania. Badanie to polega na cyklicznym rozgłaszaniu (broadcasting) w sieci informacji o swojej lokalizacji i równocześnie na podejmowaniu stosownych działań po otrzymaniu takiego informacyjnego pakietu danych. Mechanizmy te mogą być jedynym źródłem informacji o *agentach systemowych* znajdujących się w otoczeniu sieciowym, mogą także uzupełniać przekazaną *agentom systemowym* (w momencie ich inicjalizacji) wstępną statyczną listę potencjalnych lokalizacji agentów, wreszcie mogą zostać zupełnie wyłączone przy założeniu, że przekazana statyczna lista zawiera pełną informację o lokalizacji *agentów systemowych*. Należy jednak dodać, że ze względu na zastosowaną implementację polegającą na wysyłaniu pakietów danych na adres rozgłoszeniowy sieci lokalnej, mechanizm ten może być z powodzeniem wykorzystany jedynie w obrębie tej właśnie sieci. Ostatnia grupa zadań obejmuje monitorowanie stanu systemu, zarówno w aspekcie lokalnym (stan poszczególnych agentów zlokalizowanych na danym komputerze), jak również w aspekcie globalnym (wzajemne monitorowanie stanu poszczególnych *agentów systemowych*). W zakres tej grupy zadań wchodzi także działania związane z zapewnieniem integralności systemu sterowania przy uwzględnieniu możliwości wystąpienia sytuacji awaryjnych. Poszczególne zadania oraz wybrane sekwencje wykonywanych działań dla każdego z nich zostały przedstawione za pomocą UML-owego diagramu czynności na rys. 2.

Trzecim aspektem dotyczącym rozproszonego systemu sterowania produkcją i omówionym w niniejszej pracy są zagadnienia dotyczące przyznawania uprawnień decyzyjnych.

4. PRYZNAWANIE UPRAWNIEN DECYZYJNYCH

Podejmowanie decyzji dotyczących przydzielania zasobów do realizacji określonych czynności ma w systemach zdecentralizowanych szczególne znaczenie, bowiem z racji rozproszonej struktury wiedza decydenta o stanie systemu w momencie podejmowania decyzji jest zazwyczaj ograniczona. Sytuację komplikuje dodatkowo, skądinąd pożądanym, fakt potencjalnej wielowariantowości realizowanych procesów. Dlatego niezmiernie istotne staje się opracowanie wydajnych i solidnych mechanizmów pozwalających na podejmowanie decyzji najkorzystniejszych z punktu widzenia założonych celów.

Istnieją dwie strategie podejmowania decyzji w środowiskach rozproszonych [8]. Pierwsza z nich zakłada, iż decyzje podejmowane są przez określone agenty równolegle. Podejście takie oznacza, że agenty, które aktualnie nie są zaangażowane w realizację żadnych czynności, starają się równocześnie podejmować decyzje dotyczące swych dalszych działań. Z kolei realizacja określonych czynności wiąże się zazwyczaj z potrzebą rezerwacji wymaganych do jej realizacji zasobów. Równoległość procesów podejmowania decyzji w prosty sposób prowadzi do powstania cyklu wzajemnych oczekiwań zasobowych czyli blokad. Dlatego też korzystniejsza w omawianym zagadnieniu może okazać się druga strategia podejmowania decyzji polegająca na sekwencyjnym przekazywaniu uprawnień decyzyjnych agentom biorącym udział w procesach decyzyjnych. Podejście takie oznacza, że tylko jeden spośród agentów niezaangażowanych w danej chwili w realizację żadnych czynności będzie dokonywał wyboru czynności do realizacji, co w konsekwencji zabezpiecza system przed łatwym sposobem

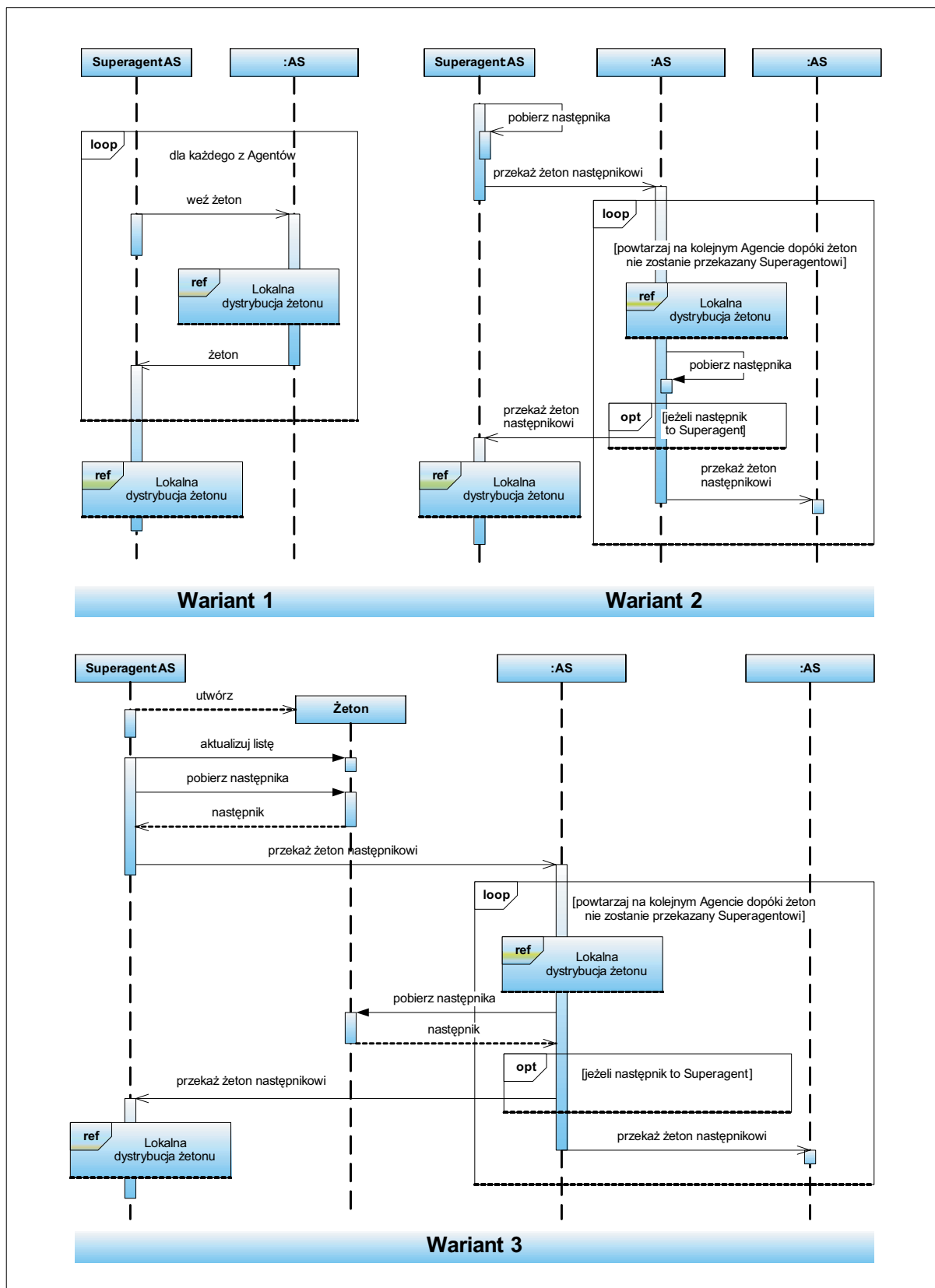
wprowadzenia systemu w stan blokady choć możliwości ich powstania całkowicie nie eliminuje.

W rozważanym systemie sterowania AIM zastosowano sekwencyjną strategię przyznawania uprawnień decyzyjnych. Proces przekazywania uprawnień przebiega tu dwustopniowo. W pierwszej fazie uprawnienia decyzyjne w postaci tzw. żetonu decyzyjnego otrzymuje *agent systemowy*, który w kolejnym etapie dystrybuje je lokalnie (tj. w obrębie „swojego” komputera) pomiędzy docelowymi decydentami. Funkcje decydentów w systemie AIM mogą sprawować, w zależności od zastosowanego podejścia: wyłącznie *agenty wykonawcze*, wyłącznie *agenty przedmiotowe* lub równocześnie agenty obydwu typów.

Po zakończeniu realizacji procesów decyzyjnych *agent systemowy* zwraca żeton lub przekazuje go dalej zgodnie z przyjętym sposobem dystrybucji żetonu. Ważną kwestią warunkującą poprawne działanie mechanizmu przyznawania uprawnień decyzyjnych jest zapewnienie istnienia w konkretnej chwili jednego i tylko jednego żetonu decyzyjnego w systemie. W tym celu spośród *agentów systemowych* powołany zostaje jeden (tzw. superagent), który poza standardowymi zadaniami *agenta systemowego* pełni dodatkowo funkcję nadzorcy koordynującego proces dystrybucji żetonu.

Istotny aspekt rozważanego zagadnienia stanowi sposób dystrybucji żetonu decyzyjnego pomiędzy *agentami systemowymi*. Jedno z możliwych rozwiązań (wariant 1) zakłada, iż każdorazowo po otrzymaniu żetonu i przeprowadzeniu lokalnych procesów decyzyjnych *agent systemowy* zwraca żeton superagentowi. W podejściu tym porządek, w jakim poszczególne agenty otrzymują żeton, znany jest tylko superagentowi, który w razie potrzeby (nowy agent w systemie, awaria istniejącego agenta) może w łatwy sposób dokonać jego modyfikacji. Drugim sposobem realizacji obiegu żetonu decyzyjnego (wariant 2) jest przekazywanie go bezpośrednio pomiędzy kolejnymi *agentami systemowymi* tworzącymi swego rodzaju logiczny pierścień. W tej sytuacji superagent w ciągu pełnego cyklu obiegu żetonu decyzyjnego otrzymuje ów żeton tylko raz. W porównaniu z wariantem pierwszym, czas pełnego obiegu żetonu pomiędzy *agentami systemowymi* jest skrócony, jednakże jakkolwiek zmiana porządku agentów w pierścieniu jest w tym przypadku trudniejsza w realizacji, wymaga bowiem bezpośredniej modyfikacji stosownych informacji zawartych w obrębie określonych *agentów systemowych*. Wreszcie trzecia możliwość (wariant 3) zakłada, że lista agentów definiująca porządek, w jakim kolejne *agenty systemowe* otrzymają żeton decyzyjny, stanowi integralną część żetonu i jest dystrybuowana wraz z nim. Każdy z *agentów systemowych* po zakończeniu realizacji lokalnych procesów decyzyjnych odnajduje na wspomnianej liście swojego następnika, któremu przekazuje żeton. Rozwiązanie to wydaje się być najkorzystniejszym z wymienionych. Poprzez odpowiedni porządek agentów na liście zawartej w żetonie decyzyjnym można w łatwy sposób zrealizować każdy z pierwszych dwóch wariantów obiegu żetonu. Jednocześnie jakiegokolwiek zmiany w liczbie aktywnych *agentów systemowych* mogą równie łatwo zostać skorygowane przez superagenta poprzez stosowną modyfikację wspomnianej listy. Rysunek 3 przedstawia UML-owe diagramy sekwencji obrazujące wszystkie trzy omówione warianty realizacji obiegu żetonu decyzyjnego.

Warto zauważyć, że analogiczne rozumowanie można także przeprowadzić rozważając lokalną dystrybucję żetonu decyzyjnego pomiędzy docelowymi decydentami znajdującymi się w bezpośrednim otoczeniu *agenta systemowego* (na tym samym komputerze).



Rys. 3. Warianty realizacji obiegu żetonu decyzyjnego

5. PODSUMOWANIE

Przedstawione w niniejszej pracy modele obrazują jedynie pewien wycinek zagadnień spotykanych w problematyce rozproszonych systemów sterowania produkcją. Efektywna realizacja takich systemów wymaga jednakże kompleksowego spojrzenia na wszystkie aspekty związane z ich budową i funkcjonowaniem. Nieunikniona staje się więc potrzeba opracowania kompletu modeli pozwalających na szczegółową analizę projektowanego systemu jeszcze przed rozpoczęciem jego implementacji. Zastosowany w przedstawionych przykładach język modelowania UML, wykorzystujący do opisu struktury i działania złożonego systemu różnego typu diagramy, stanowi właściwą bazę do przygotowywania takich modeli.

6. LITERATURA

- [1] BOOCH G., RUMBAUGH J., JACOBSON I., UML przewodnik użytkownika. WNT, 2001.
- [2] CYKLIS J., PIERZCHAŁA W., Modelowanie procesów dyskretnych w elastycznych systemach produkcyjnych. Zeszyty Naukowe Politechniki Krakowskiej, z. 77, nr 3, Kraków 1995.
- [3] CYKLIS J., SŁOTA A., Obiektowo-obszerniealna sieć Petriego w zastosowaniu do modelowania elastycznych systemów wytwarzania. Zeszyty Naukowe Politechniki Śląskiej, Seria: Automatyka, z.124, Gliwice 1998, s. 31-41.
- [4] DICESARE F., HARHALAKIS G., PROTH J.M., SILVA M., VERNADAT F.B., Practice of Petri Nets in Manufacturing. Chapman & Hall, London 1993.
- [5] FINDEISEN W. (red.), Analiza systemowa – podstawy i metodologia. PWN, 1985.
- [6] HEIKKILA T., KOLLINGBAUM M., VALCKENAERS P., BLUEMINK G-J., Man-Age: an Agent Architecture for Manufacturing Control. Proc. of the 2nd Int. Workshop on Intelligent Manufacturing Systems, Leuven 1999, s. 127-136.
- [7] ODELL J., PARUNAK H.V.D., BAUER B., Representing Agent Interaction Protocols in UML. Proc. of Agent-Oriented Software Engineering, 2000, s. 121-140.
- [8] ZAJĄC J., Rozproszone sterowanie zautomatyzowanymi systemami wytwarzania. Monografia 288, Seria Mechanika. Wydawnictwo Politechniki Krakowskiej, Kraków 2003.
- [9] ZAJĄC J., CHWAJOŁ G., KMIECIK A., Integracja projektowania procesów i sterowania produkcją w zautomatyzowanych systemach wytwarzania. Pomiary Automatyka Robotyka, 2/2007.

Praca naukowa finansowana ze środków na naukę w latach 2006-2008 jako projekt badawczy 2394/T02/2006/31 pt. „Zintegrowany system projektowania procesów i sterowania produkcją”.