

Kamil ROGALIŃSKI¹
Andrzej BARTOSZEWICZ²
Przemysław IGNACIUK³

APLIKACJA DO BADANIA DYNAMIKI PROCESÓW LOGISTYCZNYCH

W pracy przedstawiono aplikację służącą do symulacji przebiegu procesów logistycznych. Aplikacja umożliwia testowanie zarówno klasycznych jak i nowo projektowanych algorytmów sterowania w warunkach zmiennego popytu oraz przy różnych parametrach modelu. Program ma własną bazę danych, w której przechowuje informacje o towarach i dostawcach. Został też wyposażony w interfejs webowy, dzięki któremu jest kompatybilny ze wszystkimi systemami operacyjnymi i dostępny z każdego komputera na świecie. Może być stosowany do optymalizacji procesu zarządzania zasobami.

SOFTWARE TOOL FOR TESTING OF DYNAMIC LOGISTIC PROCESSES

In this paper, we introduce an application simulating logistic processes in periodic-review inventory systems. The application allows its user to test both classical supply policies and recently developed algorithms, in the presence of variable demand and different model parameters. This program is equipped with an integrated database system which holds information on commodities and suppliers. Furthermore, it is controlled by a web interface which makes it compatible with all operating systems and accessible from every computer in the world. The presented application may be used to optimize inventory control systems.

1. WSTĘP

Już w czasach starożytnych wraz z narodzinami wojsk pojawiła się potrzeba ich przemieszczenia, kwaterowania, a także zaopatrywania w sprzęt, broń, żywność itp. Wraz z rozwojem gospodarki i handlu do kluczowych zadań logistyki dołączyło zarządzanie przepływami rzeczowymi w ramach przedsiębiorstwa i między przedsiębiorstwami oraz związany bezpośrednio z nimi przepływ informacji w taki sposób, aby zaspokoić wymagania klienta. Aby sprostać stawianym wyzwaniom niezbędne jest opracowanie i

¹ Politechnika Łódzka, Wydział Elektrotechniki, Elektroniki, Informatyki i Automatyki; 90-924 Łódź; ul. Bohdana Stefanowskiego 18/22. Email: kamilrogalinski@gmail.com

² Politechnika Łódzka, Wydział Elektrotechniki, Elektroniki, Informatyki i Automatyki; 90-924 Łódź; ul. Bohdana Stefanowskiego 18/22. Email: andrzej.bartoszewicz@p.lodz.pl

³ Politechnika Łódzka, Wydział Elektrotechniki, Elektroniki, Informatyki i Automatyki; 90-924 Łódź; ul. Bohdana Stefanowskiego 18/22. Email: przemyslaw.ignaciuk@p.lodz.pl

ciągła optymalizacja strategii zarządzania zasobami [1 – 7], a do tego potrzebne są niezbędne narzędzia. Niniejsza praca poświęcona jest prezentacji jednego z takich narzędzi – opracowanej w Instytucie Automatyki Politechniki Łódzkiej aplikacji umożliwiającej badanie dynamiki procesów logistycznych.

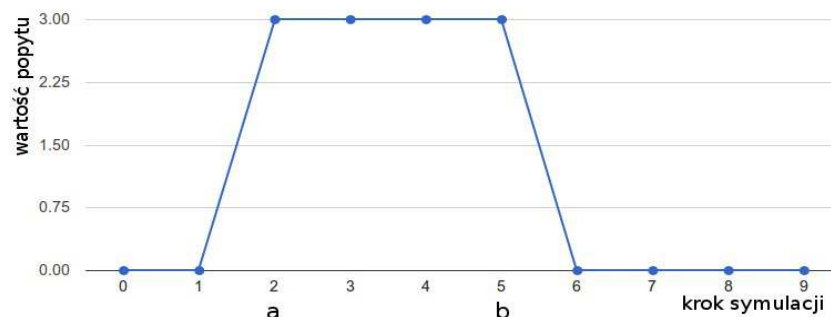
2. FUNKCJONALNOŚĆ APLIKACJI

Głównym zadaniem aplikacji jest generowanie listy zamówień na towar na podstawie zadanej funkcji popytu z uwzględnieniem wybranych parametrów modelu.

Dostępne parametry modelu to:

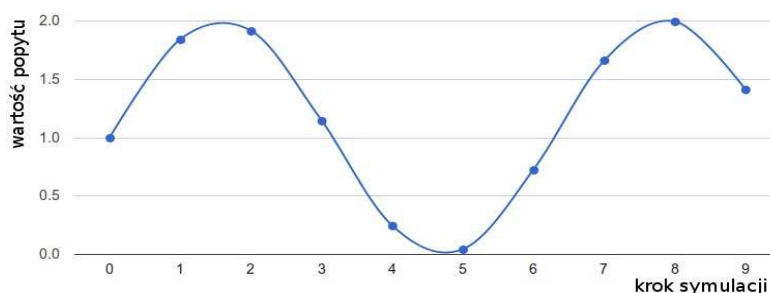
- a) Czas symulacji wyrażany liczbą całkowitą z przedziału [1, 6500].
- b) Możliwość późniejszej realizacji zamówienia (backorders) – ta wartość ma bezpośredni wpływ na przebieg symulacji w przypadku kiedy wartość funkcji popytu przewyższa ilość towaru dostępną w magazynie. Zamówienia, które nie zostaną zrealizowane mogą zostać utracone lub przełożone do realizacji w dniu następnym. W stworzonej aplikacji istnieje możliwość określenia wartością procentową ilości zamówień, które mają zostać utracone.
- c) Czas realizacji zamówienia – jest to czas jaki upływa od momentu złożenia zamówienia do czasu fizycznego przybycia towaru do magazynu. Istnieją trzy sposoby określenia tego czasu:
 - Wykorzystanie informacji przechowywanej w bazie danych. Dla każdego zdefiniowanego dostawcy oraz towaru określony jest średni czas realizacji zamówienia. Będzie on stały dla każdego złożonego zamówienia przez cały okres symulacji.
 - Określenie wartością stałą inną niż ta przechowywana w bazie danych z przedziału [0, 1000]. W tym przypadku dla każdego wygenerowanego zamówienia zastosowany będzie ten sam czas realizacji przez cały okres symulacji.
 - Określenie wartością losową z przedziału liczb całkowitych [a, b]. W tym przypadku dla każdego zamówienia zostanie wylosowany czas jego realizacji. Do generowania losowych wartości zastosowano generator liczb pseudolosowych wbudowany w architekturę języka PHP, tj. funkcję rand().⁴
- d) Popyt – jest to funkcja generująca zapotrzebowanie na towar w każdym kroku symulacji. W aplikacji zaimplementowano kilka różnych funkcji określających popyt:
 - Funkcja stała – wymaga określenia początku i końca przedziału [a, b] oraz wartości popytu. Podczas symulacji popyt w określonym przedziale czasu przyjmuje wybraną wartość, a poza przedziałem jego wartość będzie równa 0. Dla przedziału [2, 5] i wartości popytu równej 3 funkcja przyjmie takie wartości jak pokazano na rysunku 1.

⁴ rand() — Generuje losową liczbę stałoprzecinkową o rozkładzie normalnym
<<http://www.php.net/manual/pl/function.rand.php>>



Rys.1. Funkcja stała w przedziale $[a, b]$

- Funkcja liniowa określona wzorem $y = ax + b$ wymaga określenia współczynnika kierunkowego a oraz przesunięcia b .
- Popyt zmienny z przedziału $[a, b]$ – wartość popytu w każdym kroku symulacji zostanie wygenerowana za pomocą generatora liczb pseudolosowych o rozkładzie normalnym, tj. funkcji `rand()` wbudowanej w język PHP.
- Funkcja okresowa określona wzorem $y = A \sin(Bx) + C$ wymaga określenia amplitudy A , okresu funkcji B oraz przesunięcia C . Dla A, B i C równego 1 jej wykres przedstawiono na rysunku 2.



Rys.2. Funkcja okresowa

- e) Algorytm generowania zamówień – istnieje możliwość wybrania jednego lub więcej algorytmów za pomocą których generowane będą zamówienia na towar podczas symulacji. W aplikacji zaimplementowano kilka algorytmów:
- OUT (Order-up-to) - wielkość zamówienia u generowana jest na podstawie wzoru:

$$u(k) = Y_d - y(k) - WIP(k)$$

gdzie: $u(k)$ - zamówienie wygenerowane w okresie k
 Y_d - zadany poziom zasobów

$y(k)$ - ilość zasobów dostępnych w magazynie

$WIP(k)$ - ilość zasobów w zamówieniach w trakcie realizacji.

- (s, S) - algorytm generuje zamówienia do wartości S wtedy, kiedy poziom zasobów w magazynie oraz zamówień oczekujących spadnie poniżej s .
- (r, Q) - algorytm generuje zamówienia o wartości Q wtedy, kiedy stan magazynu oraz zamówień oczekujących spadnie poniżej r .
- Algorytm optymalny w sensie kwadratowego wskaźnika jakości (ang. LQ optimal) – wielkość zamówienia u generowana jest według następującego wzoru:

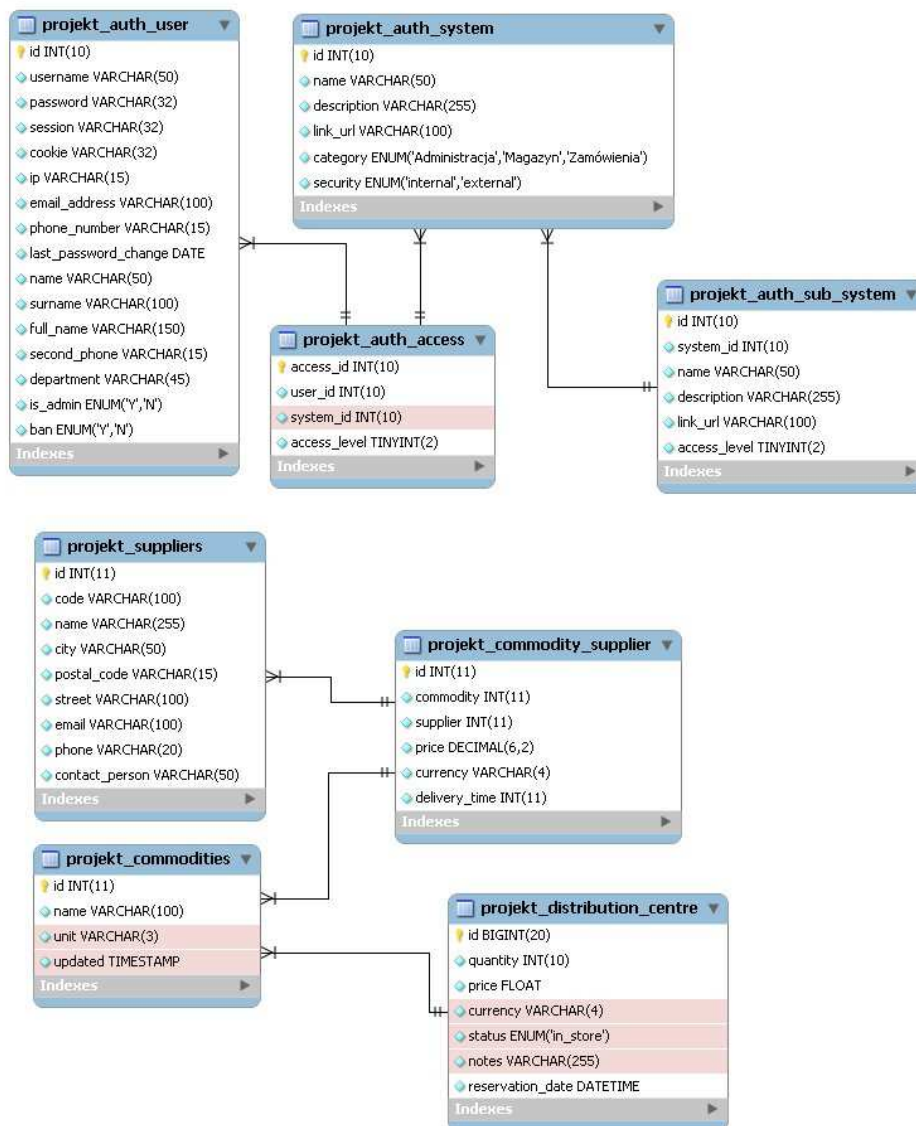
$$u(k) = B \cdot [Y_d - y(k) - WIP(k)]$$

w którym B oznacza współczynnik wzmocnienia regulatora. Ten współczynnik wynika z wag przyjętych w kwadratowym wskaźniku jakości dla kosztu sygnału sterującego i uchybu regulacji. Pozostałe parametry tego algorytmu odpowiadają parametrom opisanym powyżej przy omawianiu algorytmu OUT.

3. SPECYFIKACJA

Do stworzenia aplikacji wykorzystane zostały obiektowe techniki programistyczne. Aby uprościć procedury instalacji oraz aktualizacji aplikacji, zapewnić przejrzysty interfejs oraz mobilność, aplikacja została zaimplementowana w języku PHP ze wsparciem technologii asynchronicznej wymiany danych AJAX⁵, oraz JavaScript. Aplikacja wykorzystuje lokalną relacyjną bazę danych MySQL do przechowywania danych o dostawcach i towarach, a także informacji o użytkownikach aplikacji, zainstalowanych modułach oraz relacjach między tymi danymi. Rysunek 3 przedstawia schemat bazy danych oraz relacje między tabelami.

⁵ AJAX (ang. *Asynchronous JavaScript and XML*, asynchroniczny JavaScript i XML) , <<http://www.w3schools.com/ajax/default.asp>>



Rys.3. Baza danych

Aplikacja została wyposażona w interfejs webowy dzięki czemu można z niej korzystać bez konieczności instalacji na komputerze lokalnym, za pośrednictwem dowolnej przeglądarki internetowej obsługującej Javascript. Aby korzystać z interfejsu webowego wymagane jest aktywne połączenie z internetem.

Jeżeli komputer na którym korzystamy z programu nie jest podłączony do internetu konieczne jest wykonanie następujących kroków:

1. instalacja serwera HTTP z obsługą języka PHP,
2. instalacja i konfiguracja bazy danych MySQL,
3. skopiowanie plików aplikacji z paczki standalone.zip do głównego foldera www,
4. ustawienie zmiennych środowiskowych w pliku config.php.

Po wykonaniu tych kroków można przeprowadzać symulacje na komputerze lokalnym za pośrednictwem przeglądarki internetowej bez konieczności łączenia z internetem.

4. PRZYKŁADOWY SCENARIUSZ SYMULACYJNY

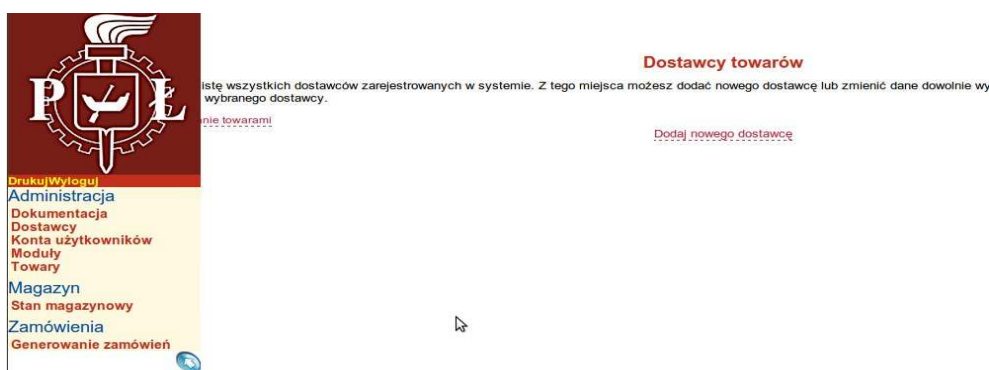
Zamawiamy jeden towar – długopisy, od 2 różnych dostawców. Towar składowany jest w magazynie o pojemności 600 sztuk. Chcemy zadbać o to, żeby ilość długopisów przechowywanych jednocześnie nie przekroczyła 400 sztuk tak, aby zachować trochę wolnego miejsca. Popyt na długopisy w ostatnim miesiącu był losowy, od 10 do 80 sztuk dziennie i zakładamy, że będzie się kształtował podobnie w kolejnych 30 dniach. Czas realizacji złożonego zamówienia u dostawcy waha się w zależności od warunków pogodowych i lokalizacji jego magazynu. Pomijamy kwestie związane z kosztami dostaw.

Zakładamy, że:

1. Czas symulacji wynosi 30 dni.
2. W przypadku gdy wartość popytu przekroczy ilość dostępną w magazynie, zamówienia zostaną zrealizowane w kolejnym dniu.
3. Wartość funkcji popytu w każdym kolejnym dniu będzie wartością losową i minimalnie wyniesie 10 a maksymalnie 80 sztuk. Wartość wyznaczana będzie za pomocą funkcji rand() generującej losowe wartości z rozkładem normalnym.
4. W symulacji badane będą 3 algorytmy sterowania przepływem towaru:
 - a) Order-up-to,
 - b) (s, S) z wartościami $s=100$ i $S=400$ sztuk,
 - c) algorytm optymalny w sensie kwadratowego wskaźnika jakości ze współczynnikiem wzmocnienia wynoszącym 0,4.
5. Towar zamawiany jest od dwóch dostawców w ilości 50% od każdego dostawcy a czasy realizacji zamówienia kształtują się następująco:
 - a) dla dostawcy pierwszego wynosi on od 2 do 5 dni, za losowanie wartości odpowiada funkcja rand();
 - b) dla dostawcy drugiego od 1 do 3 dni, także wyznaczany za pomocą funkcji rand().
6. Początkowy stan magazynu wynosi 100 sztuk, maksymalna pojemność to 600 sztuk a żądany poziom zasobów w magazynie to 400 sztuk.

4.1 Dodawanie informacji o dostawcy

W celu zdefiniowania nowego dostawcy w systemie wybieramy moduł „Dostawcy” z menu znajdującego się po lewej stronie a następnie klikamy na odnośnik „Dodaj nowego dostawcę” znajdujący się na środku ekranu. Wygląd głównego menu oraz lokalizację odnośnika przedstawia rysunek 4.



Rys.4. Dodawanie nowego dostawcy

W formularzu, który otrzymamy na następnej stronie należy uzupełnić *kod dostawcy*, *nazwę*, *miasto*, *ulicę*, *kod pocztowy*, *telefon*, *adres email* i *osobę kontaktową*. Pola wymagane to *nazwa dostawcy* i *miasto*. Przy definiowaniu nowego dostawcy nie podajemy żadnych informacji związanych z towarem gdyż od tego samego dostawcy możemy kupować różne towary i czas dostawy będzie inny dla każdego z nich.

4.2 Dodawanie towaru

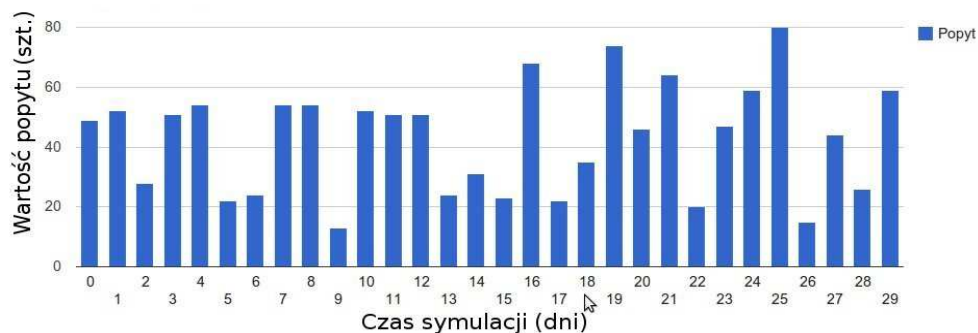
Dodawanie towaru wygląda tak samo jak w przypadku dostawcy. Wybieramy z głównego menu opcję „Towary” a następnie „Dodaj nowy towar”. W formularzu wymagane jest jedynie podanie *nazwy towaru* oraz *jednostki miary*. W następnym kroku zostaniemy poproszeni o przypisanie dostawców do nowo zapisanego towaru. Wybieramy „Dostawcę 1”, podajemy cenę, walutę oraz średni czas dostawy oraz „Dostawcę 2” i uzupełniamy wymagane pola. W tej chwili można rozpocząć symulację.

4.3 Generowanie zamówień

Aby rozpocząć symulację przepływu towaru wybieramy opcję „Generowanie zamówień” z głównego menu (rysunek 4) oraz zaznaczamy „długopis” i potwierdzamy szarym przyciskiem „Generuj zamówienia”. Następnie uzupełniamy parametry symulacji zgodnie z założeniami problemu i potwierdzamy przyciskiem „Generuj”.

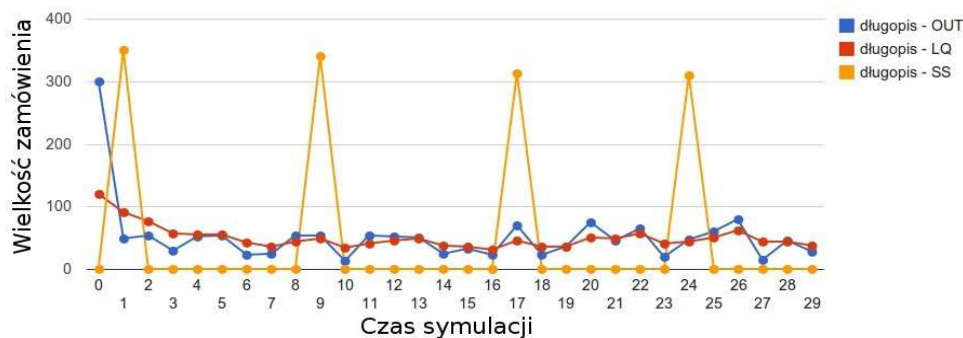
4.4 Wyniki symulacji

Pierwszym etapem symulacji jest wygenerowanie wartości funkcji popytu dla każdego kroku symulacji. Rysunek 5 przedstawia przebieg funkcji popytu dla losowych wartości uzyskanych metodą `rand()` w czasie kolejnych 30 dni.



Rys.5. Popyt w najbliższych 30 dniach

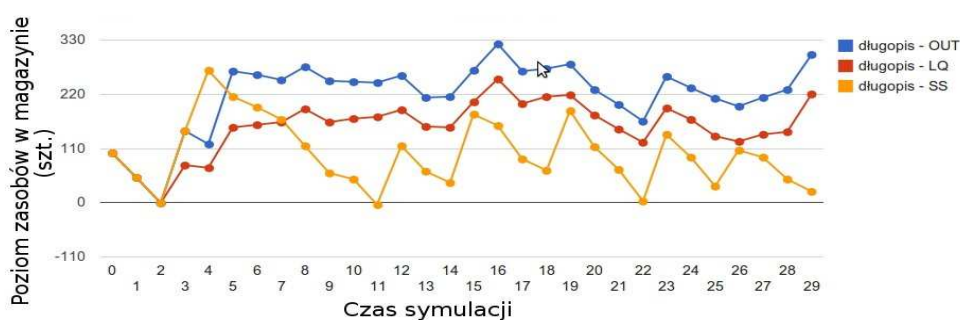
Na rysunku 6 przedstawiono zamówienia wygenerowane przez każdą z analizowanych strategii uzupełniania zasobów. Jak widać na rysunku w pierwszym dniu metoda Order-up-to oraz strategia optymalna w sensie kwadratowego wskaźnika jakości wygenerowały maksymalną dopuszczalną ilość towaru w zamówieniu, natomiast algorytm (s, S) ze względu na stan początkowy magazynu nie mniejszy od parametru $s = 100$ nie utworzył zamówienia. W kolejnych krokach symulacji generowane zamówienia zależą od poziomu zasobów w magazynie oraz ilości zamówień w trakcie realizacji.



Rys.6. Wygenerowane zamówienia

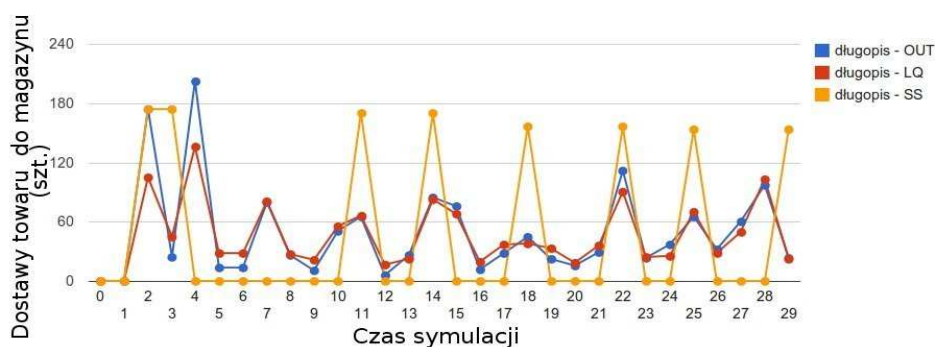
Korzystając z otrzymanych dotychczas wyników symulacji możemy sprawdzić jak będzie kształtował się stan zajętości magazynu w kolejnych 30 dniach. Jak widać na rysunku 7 metody OUT oraz LQ zapewniają taką ilość towaru, która wystarcza na pokrycie popytu

oraz nie przekracza pożądaney maksymalnej ilości towaru przechowywanego w magazynie. Mała ilość zamówień wygenerowana za pomocą algorytmu (s, S) powoduje, że kilkakrotnie brakuje towaru w magazynie, dlatego zgodnie z założeniem nadmiar popytu został oznaczony na wykresie wartościami ujemnymi i zostanie zaspokojony w kolejnym dniu.



Rys.7. Poziom zasobów w magazynie

Rysunek 8 pokazuje jak wyglądały dostawy towaru do magazynu. Na wykresie widać, że w przypadku OUT i LQ towar dostarczany był regularnie zapewniając odpowiedni poziom zasobów w magazynie. W przypadku strategii (s, S) pomimo złożonych 4 zamówień na rysunku 8 widać, że towar był dostarczony 8 razy. Dzieje się tak dlatego, gdyż towar zamawiany był od dwóch różnych dostawców i dla każdego z nich termin realizacji zamówienia był inny.



Rys.8. Realizacja zamówień

5. WNIOSKI

W pracy przedstawiono wykonaną w Instytucie Automatyki Politechniki Łódzkiej aplikację do badania zjawisk dynamicznych występujących w logistycznych łańcuchach dostaw towarów. Opisana aplikacja ma własną bazę danych, w której przechowuje informacje o towarach i dostawcach. Ponadto dzięki temu, że została wyposażona w interfejs webowy, jest kompatybilna ze wszystkimi systemami operacyjnymi i dostępna z każdego komputera na świecie. Aplikacja może być skutecznym narzędziem służącym do wspomagania optymalizacji procesu zarządzania zasobami.

Praca naukowa finansowana ze środków na naukę w latach 2010 – 2012 jako projekt badawczy nr N N514 108638 pod tytułem „Zastosowanie metod teorii regulacji do sterowania procesami logistycznymi”.

6. BIBLIOGRAFIA

- [1] Simon H. A., Holt C. C.: *The control of inventories and production rates - a survey*, Journal of the Operations Research Society of America, Vol. 2, No. 3, 1954, str. 289-301.
- [2] Ortega M., Lin L.: *Control theory applications to the production-inventory problem: a review*, International Journal of Production Research, Vol. 42, No. 11, 2004, str. 2303-2322.
- [3] Sarimveis H., Patrinos P., Tarantilis C. D., Kiranoudis C. T.: *Dynamic modeling and control of supply chain systems: a review*, Computers & Operations Research, Vol. 35, No. 11, 2008, str. 3530-3561.
- [4] Minner S.: *Multiple-supplier inventory models in supply chain management: A review*, International Journal of Production Economics, Vol. 81-82, 2003, str. 265-279.
- [5] Waters D.: *Inventory Control and Management*. West Sussex, John Wiley & Sons, 2003.
- [6] Ignaciuk P., Bartoszewicz A.: *LQ optimal sliding mode supply policy for periodic review inventory systems*. IEEE Transactions on Automatic Control, Vol. 55, No. 1, 2010, str. 269-274.
- [7] Ignaciuk P., Bartoszewicz A.: *Linear-quadratic optimal control strategy for periodic-review inventory systems*. Automatica, Vol. 46, No. 12, 2010, str. 1982-1993.