Planowanie i kontrolowanie postępu przedsięwzięć programistycznych w warunkach niepewności w MS Project

Maciej Hapke*
Instytut Informatyki, Politechnika Poznańska

Streszczenie

W artykule przedstawiono zarys metodyki i narzędzi harmonogramowania w warunkach niepewności dedykowanych dla przedsięwzięć programistycznych. Przyjęto i uzasadniono rozmyty model niepewności czasów realizacji zadań. Zaproponowano metodę analizy ryzyka przekroczenia planowanego terminu zakończenia projektu. Przedstawiono implementację w MS Project 2000.

1. Wprowadzenie

Nie ulega wątpliwości, że planowanie przedsięwzięć programistycznych jest jednym z najtrudniejszych zadań planowania. W tej dziedzinie szczególnie często zdarzają się opóźnienia, złe oszacowania kosztu i czasu, w rezultacie czego powstają produkty słabej jakości. Jest to często ściśle związane z błędami w harmonogramowaniu i monitorowaniu, w szczególności z nieuwzględnieniem nieuniknionej niepewności w planowaniu przedsięwzięć programistycznych. Ta praca chce wyjść naprzeciw trudnościom związanym z harmonogramowaniem w warunkach niepewności. Nawiązuje ona do wcześniejszej pracy [1] w której przedstawiono ideę integracji narzędzi harmonogramowania przedsięwzięć programistycznych w warunkach niepewności. Zaprezentowano w niej prototypową aplikację Fuzzy Project, w której czasy trwania poszczególnych zadań modelowane są za pomocą liczb rozmytych. Pomimo ograniczeń aplikacja ta spotkała się z

* Praca finansowana z grantu KBN 43-369/DS

zainteresowaniem dowodzącym możliwości praktycznego wykorzystania rozmytego modelu niepewności. Fuzzy Project mógł być wykorzystany w praktyce, ale tylko w aspekcie zarządzania czasem przedsięwzięcia (ang. project time management). Do zarządzania kosztem przedsięwzięcia (ang. project cost management) niezbędne było wykorzystanie innego narzędzia.

Wśród dostępnych na rynku narzędzi zarządzania przedsięwzięciami dużą popularnością cieszy się MS Project. Pakiet ten posiada wiele możliwości, dzięki którym silnie wspomaga zarządzanie w najważniejszych fazach przedsięwzięcia, np. podczas planowania i realizacji. Posiada również wygodne narzędzia analizy czasu i kosztu oraz planowania wykorzystania zasobów. Doceniając te i inne zalety oraz dużą popularność MS Project autor postanowił rozszerzyć ten pakiet o dodatkowe narzędzia m.in. moduł modelowania niepewności, dedykowane do zarządzania przedsięwzięciem programistycznym. System MS Project 2000 jako składnik rodziny aplikacji biurowych firmy Microsoft oferuje pełne wsparcie dla języka programowania Visual Basic For Application. Poprzez ten język programowania możliwy jest dostęp do prawie każdego składnika tego systemu poczynając od elementów interfejsu, a kończąc na strukturach danych przechowujących informacje dotyczące otwartych i używanych projektów.

Dzięki temu rozszerzono MS Project o możliwość definiowania rozmytych czasów realizacji zadań, zaimplementowano rozmytą procedurę propagacji niepewności poszczególnych zadań, w rezultacie której budowany jest rozmyty harmonogram zadań. Zaimplementowany zestaw narzędzi umożliwia ocenę niepewności czasu realizacji przedsięwzięcia, związaną z tym ocenę ryzyka przekroczenia terminu krytycznego, ocenę rozkładu możliwych nakładów potrzebnych na realizację projektu.

2. Rozmyty model niepewności

2.1. Dlaczego model rozmyty

Od lat toczy się dyskusja nad tym który z modeli stochastyczny czy rozmyty najbardziej nadaje się do modelowania niepewności. Nie chcąc zabierać głosu w tej dyskusji autor postanowił przyjrzeć się niektórym opiniom autorytetów [3]. Z tej niekończącej się debaty można wywnioskować, że w pewnym sensie oba modele nie konkurują ze sobą. Jeżeli istnieją dane historyczne w postaci rozkładu prawdopodobieństwa zajścia zdarzenia to najlepszym sposobem jest wykorzystanie stochastycznego modelu niepewności, który w sposób dokładny modeluje niepewność. Należy tu jednak podkreślić, że aby korzystać z takiego modelu należy posiadać reprezentatywną próbę danych historycznych. Próba reprezentatyw-

na oznacza próbę o odpowiedniej wielkości, zebraną w tych samych warunkach. Czy jest to możliwe w przypadku projektów programistycznych?

W dziedzinie produkcji oprogramowania

- projekty zwykle trwają długo,
- zadania są praktycznie niepowtarzalne, np. nie implementuje się np. po raz drugi modułu już raz zaimplementowanego w przeszłości
- w wyniku szybkiego rozwoju informatyki, nowe przedsięwzięcia są często wykonywane z wykorzystaniem nie stosowanych wcześniej technik i narzędzi,
- różnice efektywności pracy poszczególnych osób są duże,
- przepływ ludzi jest duży,

dlatego istnieje obawa, że skompletowanie danych o naturze stochastycznej nie jest łatwe jeśli nie niemożliwe.

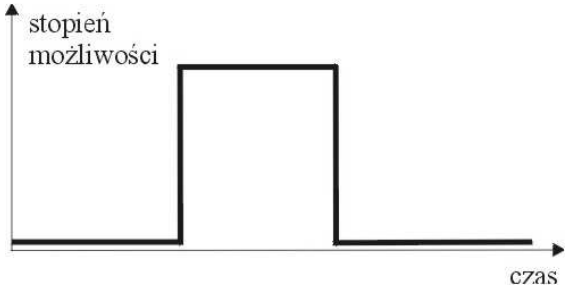
Nie oznacza to oczywiście, że czasy realizacji zadań w przedsięwzięciu programistycznym są zupełnie nieprzewidywalne. Doświadczony menadżer uwzględniając wiedzę na temat używanych technik i narzędzi budowy oprogramowania oraz doświadczenie poszczególnych projektantów i programistów jest z pewnością w stanie w przybliżeniu określić czasy wykonywania poszczególnych zadań. Może np. określić pewien interwał czasowy określający minimalny i maksymalny możliwy czas realizacji zadania. Dodatkowo może określić taki interwał czasowy, który odpowiada sytuacji najbardziej możliwej. Tego typu niepewność może być w naturalny sposób wyrażona za pomocą liczb rozmytych. Zdaniem wielu naukowców zajmujących się modelowaniem niepewności [3] rozmyty model niepewności bardziej niż stochastyczny nadaje się do wyrażania intuicji eksperta.

Brak takich narzędzi harmonogramowania, które umożliwiają intuicyjne wyrażenie niepewności skłonił autora do podjęcia prac nad nowym narzędziem.

2.2. Różne reprezentacje rozmytych czasów trwania

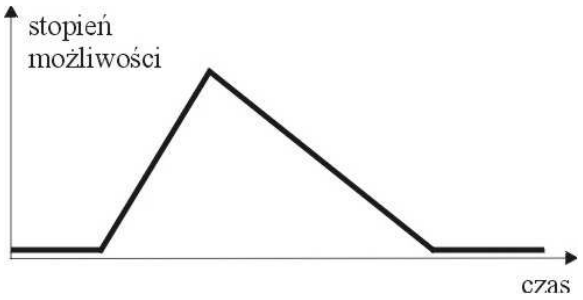
W literaturze spotyka się różne modele liczb rozmytych. Najczęściej stosowane są modele z funkcjami przynależności, które są odcinkami liniowe. W pracy proponujemy wykorzystanie trzech modeli. Na poniższych rysunkach przedstawimy i uzasadnimy wykorzystanie każdego z nich.

Jeżeli czas trwania zadania określony jest jako nie krótszy niż, nie dłuższy niż, i wszystkie wartości pomiędzy wartościami skrajnymi wydają się dla eksperta równie możliwe, to odpowiada to modelowi przedziału liczbowego.



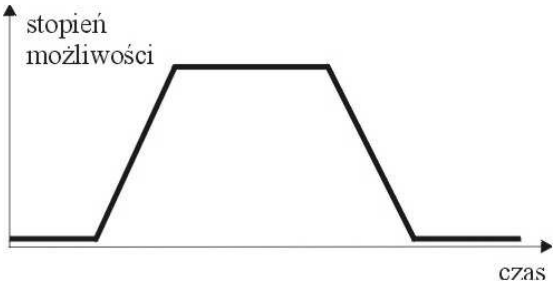
Rys.1. Przedział liczbowy jako 2-punktowy model liczby rozmytej

Jeżeli, ekspert potrafi określić wartości optymistyczną, najbardziej możliwą i pesymistyczną czasu trwania zadania, to informację taką może wyrazić w trzypunktowym modelu liczby rozmytej. Przyjęcie takiego modelu oznacza, że ekspert nie spodziewa się czasu trwania zadania krótszego od wartości optymistycznej, oraz dłuższego od wartości pesymistycznej, a najbardziej spodziewa się, wartości którą określił jako najbardziej możliwą.



Rys.2. Model 3-punktowy liczby rozmytej

W końcu ekspert może określić wartości najbardziej możliwe jako przedział liczbowy, a oprócz tego podać wartości optymistyczne i pesymistyczne. Odpowiada temu czteropunktowy model liczby rozmytej.



Rys.3. Model 4-punktowy liczby rozmytej

Poszczególne modele mogą być zrealizowane w rozszerzeniu MS Project w postaci tabelarycznej albo z wykorzystaniem następujących funkcji

- UNIFORM(x,y) dla reprezentacji przedziałowej,
- TRIANGLE(x,y,z) dla reprezentacji 3 punktowej
- TRAP(x,y,z,w) dla reprezentacji 4 punktowej.

Np. rozmyty czas wykonywania zadania d = TRAP(5, 6, 8, 10) oznacza najbardziej możliwy czas wykonania zadania pomiędzy 6 a 8 dni. Ponadto spodziewamy się, że czas ten nie będzie na pewno krótszy niż 5 dni i nie dłuższy niż 10 dni.

3. Metodyka harmonogramowania przedsięwzięć z rozmytymi czasami realizacji zadań

3.1. Metody harmonogramowania

Na pewnym etapie harmonogramowania przedsięwzięć, po podziale projektu na fazy i na zadania, po ustaleniu zależności kolejnościowych, przydzieleniu zasobów do zadań i określeniu dostępności zasobów w projekcie, zachodzi potrzeba weryfikacji planu. MS Project na bieżąco w czasie definiowania przedsięwzięcia dokonuje zmian w harmonogramie wykorzystując metodę ścieżki krytycznej (ang. Critical Path Method, CPM). Jak wiadomo metoda ta uwzględnia jedynie zależności kolejnościowe pomiędzy zadaniami. Dlatego weryfikacja planu przed ustaleniem planu odniesienia (ang. baseline) polega m.in. na sprawdzeniu wykorzystania zasobów, tzn. przeciążenia poszczególnych zasobów. Jeżeli przeciążenia występują, to taki konflikt można rozwiązać korzystając z techniki wyrównywania profilu wykorzystania zasobów (ang. resource leveling). W MS Project wyrównywanie polega na tym, że w zbiorze zadań, w którym występuje konflikt zasobowy, te zadania są opóźniane, które mają niższy priorytet lub identyfikator (opcja do wyboru). Jest to technika znana w literaturze pod nazwą heurystyk priorytetowych. Istnieje duży zbiór metod wyznaczania priorytetów zadań, i zaobserwowano, że część z nich daje krótsze od innych harmonogramy.

3.2. Propagacja niepewności

Jeżeli czasy trwania zadań obarczone są niepewnością, to w naturalny sposób wpływa to na niepewność związaną z czasem rozpoczęcia następników. W

taki sposób niepewności poszczególnych zadań kumulują się i mają wpływ na termin zakończenia całego przedsięwzięcia.

3.2.1 Model PERT

Jedną z najbardziej popularnych metod propagacji niepewności jest metoda PERT [2]. Wykorzystują ją m.in. MS Project, Primavera Project Planner i in. Zakłada się, że niepewny czas trwania zadania jest zgodny z rozkładem β. W modelu tym ekspert ma możliwość oszacowania czasu trwania w trzech wersjach: optymistycznej, najbardziej prawdopodobnej i pesymistycznej. Po dokonaniu takiego oszacowania dla każdego zadania oblicza się wartość oczekiwaną i na tej podstawie korzystając z metod harmonogramowania oblicza się oczekiwany czas zakończenia całego przedsięwzięcia i wariancję. Problem polega na tym, że jak już wyżej wspomniano, w praktyce dla wielu przedsięwzięć programistycznych poszczególne oszacowania bazują na intuicji.

3.2.2 Symulacja Monte-Carlo

Innym często stosowanym sposobem propagacji niepewności dla stochastycznego modelu niepewności jest symulacja Monte-Carlo. Polega ona na wielokrotnym (setki do tysięcy razy) uruchamianiu procedury harmonogramowania dla losowo generowanych wartości zmiennych losowych wg rozkładów prawdopodobieństwa. Wyniki są zapamiętywane i tworzą rozkład prawdopodobieństwa możliwych terminów zakończenia przedsięwzięcia. Ten sposób propagacji niepewności przy dużej ilości uruchomień (ok. 1000 razy) jest dokładny, ale nie ma to znaczenia w sytuacji, gdy niepewność jest oparta na intuicji eksperta. Ponadto wadą tego podejścia jest duża czasochłonność pojedynczej symulacji.

3.2.3 Model rozmyty

W przypadku rozmytego modelu niepewności znane są dwa podejścia do propagacji niepewności.

Tradycyjne podejście polega na transformacji oryginalnego problemu do problemu zastępczego z pewnymi wartościami wszystkich parametrów. Transformacji takiej można dokonać wybierając np. scenariusz pesymistyczny, optymistyczny lub najbardziej możliwy. Menadżer może być zainteresowany wynikami optymalizacji różnych scenariuszy. W przypadku harmonogramowania scenariusz pesymistyczny oznaczałby np., że czasy realizacji wszystkich zadań ustalane są na najdłuższym (najgorszym) możliwym poziomie. Zaletą takiego podejścia jest to, że problem zastępczy może być rozwiązywany za pomocą tradycyjnych narzędzi harmonogramowania. Wadą takiego podejścia jest to, że harmonogramy odpowiadające różnym scenariuszom mogą wymagać (i w praktyce z reguły wymagają) różnej kolejności wykonania zadań. Menadżer nie wie więc, który z tych harmonogramów powinien wybrać.

Inny sposób polega na propagacji niepewności przy wykorzystaniu arytmetyki rozmytej. Taki sposób rozwiązania rozmytego problemu harmonogramowania został zaproponowany w [4], a zaimplementowany w postaci systemu Fuzzy Project [1]. Zaproponowane przez autorów podejście daje w wyniku harmonogram, w którym czasy rozpoczęcia i zakończenia poszczególnych zadań są rozmyte, a kolejność realizacji zadań jest jednoznacznie ustalona i możliwa do zrealizowania przy wszystkich możliwych czasach realizacji zadań elementarnych.

W modelu rozmytym czas zakończenia projektu oblicza się w jednym przebiegu. Jest to możliwe dzięki prostym operacjom arytmetycznym na liczbach rozmytych. Poniżej przedstawiono dwie operacje arytmetyczne dodawania i maksimum, które są wykorzystywane w rozmytej procedurze harmonogramowania.

dodawanie

$TRAP(a_1, a_2, a_3, a_4) + TRAP(b_1, b_2, b_3, b_4) = TRAP(a_1+b_1, a_2+b_2, a_3+b_3, a_4+b_4)$

maksimum

$$MAX(TRAP(a_1, a_2, a_3, a_4), TRAP(b_1, b_2, b_3, b_4)) = TRAP(MAX(a_1,b_1), MAX(a_2,b_2), MAX(a_3,b_3), MAX(a_4,b_4))$$

Propagowana w ten sposób niepewność jest kumulowana i na każdym etapie przedsięwzięcia możliwa do wyznaczenia w prosty sposób. Wartości rozmytych czasów zakończenia i rozpoczęcia zadania lub grupy zadań są istotną informacją dla menadżera projektu. Informują o rozrzucie możliwości różnych scenariuszy czasów realizacji odpowiednich zadań.

Procedury harmonogramowania dostępne w MS Project operują jedynie na liczbach rzeczywistych, a ponieważ w proponowanej metodzie nie ma przejścia z modelu rozmytego do deterministycznego, dlatego należało zaimplementować nowe procedury harmonogramowania operujące na zmiennych rozmytych. Wyniki (harmonogram – czasy rozpoczęcia i zakończenia, koszty) rozpatruje się konsekwentnie w kategoriach przyjętego modelu liczb rozmytych. Przejście od definicji projektu do harmonogramu dokonywane jest na życzenie kierownika w sposób bardzo prosty. Intuicyjny model, szybkość obliczeń i prostota narzędzi dają szanse na łatwe wdrożenie.

3.3. Narzędzia kontroli harmonogramu

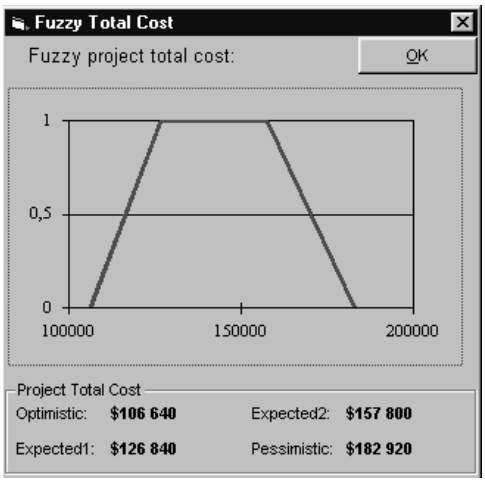
W dalszej części przedstawione zostaną narzędzia służące do kontroli harmonogramu w charakterystycznych momentach przedsięwzięcia. Narzędzia te będą korzystać z informacji zgromadzonych w wyniku uruchomienia rozmytej procedury harmonogramowania.

3.3.1 Czas trwania zadania lub zbioru zadań

Menadżer projektu może potrzebować informacji nt. rozmytego czasu trwania lub rozmytego terminu zakończenia zadania lub grupy wybranych zadań. Informacja taka może być przydatna w sytuacji potrzeby oceny stopnia niepewności ukończenia ważnych zadań lub całości projektu. Dalej może to być wykorzystane do analizy ryzyka przekroczenia terminu krytycznego (patrz 3.4).

3.3.2 Całkowity rozmyty koszt przedsięwzięcia

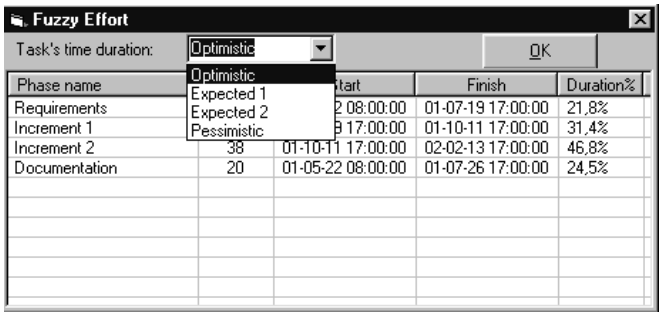
Ten moduł może być bardzo przydatny podczas szacowania całkowitego kosztu przedsięwzięcia. Podobnie jak w przypadku terminu zakończenia istotne są tu nie tylko najbardziej możliwe, ale wszystkie możliwe scenariusze. Rozmyty wynik daje szerszy pogląd na spodziewane całkowite koszty i pozwala zaplanować potrzebne rezerwy.



Rys.4. Rozmyty koszt przedsięwzięcia

3.3.3 Procentowy rozkład nakładu pracy

Ten moduł oblicza procentowy rozkład nakładu pracy na wykonanie poszczególnych faz projektu. Narzędzie to może pomóc menadżerowi w sprawdzeniu poprawności oszacowania rozkładów czasu na poszczególne fazy. Jeżeli pomiędzy fazami zachodzą zależności kolejnościowe zgodne z modelem kaskadowym to wówczas procentowy nakład czasu sumuje się do 100%. W przypadku, gdy poszczególne fazy mogą być realizowane równolegle wówczas powyższa własność nie zachodzi. Moduł umożliwia obserwowanie nakładu czasu zarówno dla optymistycznego, oczekiwanego1, oczekiwanego2 i pesymistycznego uszeregowania. Na Rys.5. przedstawiono przykładowe rozwiązanie.



Rys.5. Rozkład nakładu pracy w poszczególnych fazach projektu

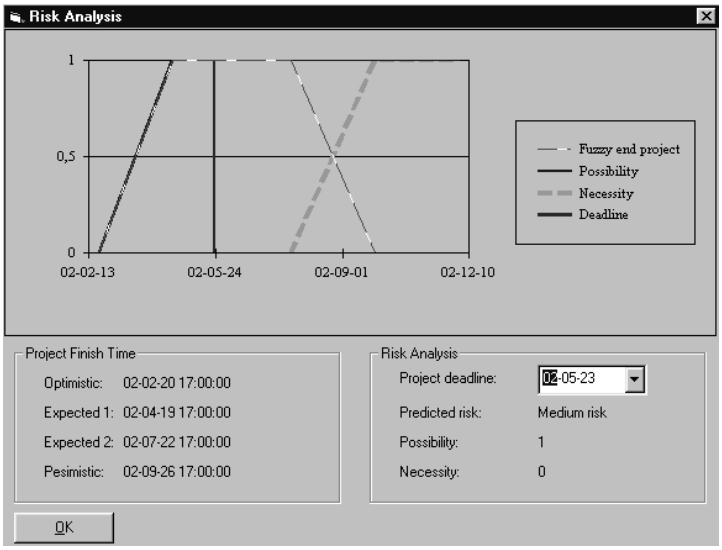
3.4. Analiza ryzyka przekroczenia terminu krytycznego projektu

Jedną z pierwszych, a zarazem najważniejszych, faz zarządzania ryzykiem jest identyfikacja zdarzeń, które mogą mieć wpływ na realizację celów projektu. W przypadku przedsięwzięć o ściśle określonym terminie zakończenia prac najważniejszym zagrożeniem jest niekorzystny krytyczny termin zakończenia projektu. Przekroczenie takiego terminu może mieć ogromny wpływ na sytuację finansową firmy realizującej zadanie.

Kolejnym krokiem w zarządzaniu ryzykiem jest dogłębna analiza wszystkich potencjalnych zagrożeń. W klasycznym podejściu należy zidentyfikować prawdopodobieństwo wystąpienia niekorzystnych czynników. W przypadku przedsięwzięć programistycznych jest to zwykle bardzo trudne. Brak tych danych powoduje, że analiza ryzyka jest niemożliwa. W przypadku rozpatrywania terminu krytycznego projektu realizowanego wg rozmytego harmonogramu można jednak posłużyć się pewnymi miarami teorii zbiorów rozmytych miarami możliwości i konieczności.

Wykorzystując definicje miary konieczności i możliwości można zbudować procedurę obliczającą ryzyko przekroczenia terminu krytycznego projektu dla poniższych danych wejściowych.

- Rozmyty termin zakończenia projektu,
- Sztywny termin krytyczny zakończenia projektu - zwykle narzucany przez klienta i określony w umowie realizacji danej aplikacji,
- Predefiniowane przedziały ryzyka: bardzo wysokie, wysokie, średnie, niskie, bardzo niskie.



Rys.6. Wygląd modułu oceny ryzyka przekroczenia terminu krytycznego

4. Podsumowanie

Praca prezentuje metodologię harmonogramowania przedsięwzięć w warunkach niepewności modelowanej za pomocą liczb rozmytych. Proponuje praktyczne narzędzia wspomagające menadżera w zarządzaniu przedsięwzięciem programistycznym, zrealizowane w środowisku często wykorzystywanego MS Project. Narzędzia te wzbogacają jego funkcjonalność o możliwość definiowania rozmytych czasów realizacji zadań, propagowania niepewności z wykorzystaniem arytmetyki rozmytej, kontroli czasu zakończenia wybranych zadań harmonogramu, oceny rozmytego kosztu całkowitego projektu i analizę ryzyka przekroczenia terminu krytycznego. Zaletą implementacji metodyki w środowisku MS Project jest możliwość natychmiastowego wykorzystania, dla dotychczasowych użytkowników tego pakietu. W najbliższej przyszłości autor zamierza dokonać ewaluacji narzędzi i pragnie do tego zaprosić zainteresowane przedsiębiorstwa informatyczne.

Podziękowania

Autor pragnie podziękować mgr Tomaszowi Felczyńskiemu, oraz mgr Przemysławowi Kulczakowi i mgr Pawłowi Słomczewskiemu, których promotorem miał przyjemność być w latach 2001 i 2000 za implementacje metodologii w środowisku MS Project.

Bibliografia

[1]. M. Hapke, A. Jaszkiewicz, P. Kominek (1999). Zintegrowane narzędzia harmonogramowania przedsięwzięć programistycznych w warunkach niepewności. Materiały I Krajowej Konferencji Inżynierii Oprogramowania, Kazimierz Dolny 11-13.10.1999, Informatyka Stosowana S4/99, Politechnika Lubelska, 65-76.

[2]. S.E. Elmaghraby, On the expected duration of PERT type networks, Management Science 5, 299-306, 1967.

[3]. F.A. Loostma, Stochastic and fuzzy PERT, *European Journal of Operational Research*, 43 (1989), 174-183.

[4]. M. Hapke, R. Słowiński, Fuzzy priority heuristics for project scheduling, Fuzzy Sets and Systems 83, 291-299, 1996.

[5]. T. Felczyński, Zarządzanie przedsięwzięciem programistycznym w warunkach niepewności w środowisku MS Project 2000, praca dyplomowa magisterska, Poznań 2001.