

prof. dr hab. inż. Andrzej Obuchowicz

Zielona Góra, 19 luty 2023

Uniwersytet Zielonogórski

Instytut Sterowania i Systemów Informatycznych

e-mail: a.obuchowicz@issi.uz.zgora.pl

## **RECENZJA**

rozprawy doktorskiej Pana mgr inż. Leszka Ciopińskiego

*Synteza samoadaptacyjnych systemów czasu rzeczywistego z pomocą  
rozwojowego programowania genetycznego*

opracowana na zlecenie

Rady Dyscypliny Naukowej Informatyka Techniczna i Telekomunikacja  
Politechniki Warszawskiej

### **1. Problem badawczy i jego znaczenie**

Nieustanny wzrost, niemalże wykładniczy, aplikacji systemów budowanych realizowanych w realiach czasu rzeczywistego w otaczających nas produktach technologicznych wymusza na badaczach poszukiwanie coraz to bardziej zaawansowanych metod syntezy oprogramowania tychże systemów. Szczególnie istotnym obszarem stanowią zastosowania systemów w układach izolowanych energetycznie, bez stałego dostępu do zewnętrznych źródeł energii, bazujące głównie, lub jedynie, na mobilnych magazynach energii. W takich przypadkach istotnym elementem jest optymalna gospodarka energią. Praca w realiach czasu rzeczywistego, w warunkach zmiennych, trudnych do przewidzenia, wymaga nadania systemowi cech elastyczności, samoadaptacji do zmiennych wymagań czasowo-proceduralnych. Kreacja modeli syntezy oprogramowania dla takich systemów staje się niezwykle inte-

resującym zagadnieniem. Wobec powyższych faktów cele i teza pracy przedstawione w trzecim rozdziale rozprawy doktorskiej pana mgr inż. Leszka Ciopińskiego uznać należy za słuszne i zasadne.

## **2. Wkład autora**

Zasadniczy oryginalny wkład doktoranta to opracowanie metody syntezy samoadaptacyjnych systemów czasu rzeczywistego spełniających zadane wymagania czasu realizacji i poprawiające parametry oszczędnościowe pobieranej energii. W odróżnieniu od dotychczasowych propozycji, w których nacisk położono na gwarancję wymaganego terminu, w związku z tym harmonogramowanie zadań uwzględniało najbardziej niekorzystne warunki realizacji zadań, a co za tym idzie, są to rozwiązania wysoko energetyczne, w proponowanym rozwiązaniu harmonogram zadań dostosowano do najbardziej prawdopodobnych warunków. W przypadku zajścia sytuacji niekorzystnej, system dynamicznie poprawia przydział zadań na zasoby w celu uniknięcia zagrożenia naruszenia terminu realizacji i przy możliwie najniższym koszcie energetycznym.

Uzyskanie powyższego modelu wymagało opracowania wielu szczegółowych zagadnień, wśród nich wyróżnić należy:

- propozycja miar jakości do oceny samoadaptacyjności uszeregowania czasu rzeczywistego (SURT) i samooptymalizacji poboru energii (SPE), oraz ich wersji uproszczonych;
- opracowanie, dla rozważanej klasy zagadnień, reprezentacji osobnika w rozwojowym algorytmie genetycznym i dopasowanie do jego postaci operacji rekombinacji i mutacji;
- przeprowadzenie badań eksperymentalnych potwierdzających skuteczność proponowanego narzędzia.

## **3. Poprawność**

Opiniowana praca, zawarta na 137 stronach, składa się z 7 rozdziałów, w tym wstępu i podsumowania, uzupełnionych streszczeniami w języku angielskim i polskim,

spisem symboli i oznaczeń, oraz bibliografią zawierającą 113 pozycji cytowanej literatury, zawierającej wyczerpujący zbiór publikacji obejmujący stan badań prowadzonych w obszarze zainteresowania pracy.

Zawartość wstępu, stanowiącego rozdział pierwszy, stanowi w pewnym sensie motywację podjęcia zagadnienia, uwypuklając istotność systemów samoadaptacyjnych we współczesnych rozwiązaniach technologicznych. W rozdziale drugim doktorant szczegółowo rozwinął podjętą myśl wstępu przedstawiając przegląd stosowanych metod samoadaptacyjnych, szczególnie w systemach czasu rzeczywistego. Cele oraz teza pracy została zdefiniowana w rozdziale trzecim. W szczególności wskazano w nim na cechy rozważanej klasy systemów, oczekiwanych własności budowanego modelu, a także tok postępowania prowadzącego do osiągnięcia przedstawionych celów i udowodnienia postawionej tezy. Podstawowe narzędzie użyte do szeregowania zadań – rozwojowe programowanie genetyczne – zostało opisane w rozdziale czwartym. W szczególności przedstawiono technikę kodowania osobnika populacji, powiązane z nim operatory genetyczne: mutację i rekombinację, a także wskazano metodę selekcji. Opis pełnej metody syntezy samoadaptacyjnego oprogramowania systemów wbudowanych czasu rzeczywistego z wykorzystaniem rozwojowego programowania genetycznego zawarto w rozdziale piątym. W rozdziale szóstym przedstawiono badania eksperymentalne mające na celu wykazanie skuteczności proponowanej metody w odniesieniu do zaproponowanych wcześniej kryteriów w zagadnieniach szeregowania zadań z ograniczoną dostępnością i silnymi ograniczeniami. Rozdział siódmy zawiera podsumowanie całej rozprawy.

Z merytorycznego punktu widzenia przeprowadzone badania i analizy są właściwe i poprawnie zrealizowane. Stwierdzenia w większości są uzasadnione, bądź podane są odnośniki do literatury, gdzie odpowiednie uzasadnienie można odnaleźć. Tok głównej myśli rozprawy jest właściwy i dobrze przemyślany. Na szczególne wyróżnienie zasługuje przejrzystość przeprowadzanych rozważań i analiz, jak również niezwykła dbałość o postać edycyjną pracy.

Jednakże poniżej podaję kilka uwag merytorycznych.

1. W rozdziale drugim wśród omawianych metod występują także metody, których doktorant jest współautorem – wskazują na to odwołania do publikacji doktoranta. Należałoby wyróżnić w tekście fakt, że są to prace zawierające

wyniki omawiane dalej w rozprawie. Bez tej informacji czytelnik ma wrażenie, że prace te zawierają tylko pewne poprzedzające rezultaty nie objęte w rozprawie.

2. Rys. 4.1 i komentarz do niego. Algorytmy ewolucyjne można podzielić na dwie klasy: przeszukiwania genotypowe i fenotypowe. W pierwszym przypadku operujemy na dwóch przestrzeniach, przestrzeni fenotypowej, na której określona jest optymalizowana funkcja dopasowania, i przestrzeni genotypowej, na której przebiega proces ewolucyjny, czyli określone są operacje reprodukcji, mutacji, rekombinacji, czy sukcesji. W tym przypadku genotyp zawiera zakodowaną wersję fenotypu (cech), na podstawie których oceniany jest osobnik. Trzeba nadmienić, że kodowanie fenotypu w genotyp zwykle jest nieizomorficzne i zwykle stratne. Najbardziej znanymi reprezentantami tego typu przeszukiwań są algorytmy genetyczne i programowanie genetyczne. W drugim przypadku nie kodujemy fenotypu, tylko ocena osobnika i proces ewolucyjny realizowany jest bezpośrednio na fenotypach. Reprezentantami tego typu przeszukiwań są np. programowanie ewolucyjne, czy strategie ewolucyjne. Rysunek 4.1b sugeruje, że mamy do czynienia z przeszukiwaniem genotypowym, a rysunek 4.1a przeszukiwaniem fenotypowym, do której to klasy algorytmy genetyczne nie należą.
3. Str. 42 trzeci punkt listy punktowanej. Nie rozumiem dlaczego mamy unikać przypadku uzyskania lepszego osobnika jako wyniku przypadkowego losowania, tylko musi to być efekt krzyżowania.
4. Rozdział 4. W opisie proponowanego algorytmu rozwojowego programowania genetycznego brakuje ogólnego schematu całego procesu. Np. niejasne jest miejsce selekcji. Czy, w przypadku  $n$ -elementowej populacji, selekcji turniejowej rzędu  $m$  używamy w celu wygenerowania  $n$  osobników rodzicielskich (zwycięzców  $n$  turniejów grup liczebności  $m$ ), a następnie po mutacjach i krzyżowaniach otrzymamy  $n$ -elementową populację nowej generacji? Czy z aktualnej  $n$ -elementowej populacji wybieramy metodą turniejową  $k$ -elementową podpopulację rodzicielską, która jest zastępowana przez utworzonych z niej za pomocą mutacji i krzyżowań potomków, bądź wybieramy nową  $n$ -elementową populację nowej generacji jako najlepszych osobników  $n + k$ -elementowej populacji poprzedniej generacji uzupełnionej  $k$  potomkami? Opcji jest wiele. Nie

podano jaką użyto w pracy i dlaczego.

5. Wybór metody selekcji w postaci selekcji turniejowej uzasadniono powołując się na badania opublikowane w pracy [47], której doktorant jest współautorem. Dla pełności raportu badawczego, opis tych badań powinien znaleźć się w rozprawie, chociażby w postaci dodatku.
6. Definicje od 5.3 do 5.7 bardzo zbliżone są do założeń znanej z obszaru badań operacyjnych probabilistycznej metody planowania i kontroli projektu z analizą kosztową (PERT-COST, ang. Project Evaluation and Review Technique), czy elementy tej metody były wykorzystywane w budowie modelu? np. przy oszacowaniu maksymalnych opóźnień realizacji danego zadania na danym zasobie nie zagrażającym końcowemu reżimowi czasowemu.
7. Pojęcie definicji używane jest do pełnego określenia pojęcia. Czy definicje 5.9 i 5.10 spełniają ten warunek, tzn. czy możliwe są inne wyrażenia definiujące  $j^{SURT}$  i  $j^{SPE}$  niż wyrażenia (5.7) i (5.11)? Jeżeli nie, to wyrażenia (5.7) i (5.11) powinny być zawarte odpowiednio w definicjach 5.9 i 5.10.
8. Podobnie rozumując jak wyżej wyrażenie (5.2) powinno być objęte definicją 5.14, (5.3) powinno być objęte definicją 5.15 i wyrażenie (5.4) powinno być objęte definicją 5.16.
9. Strona 62. Słuszne jest zdanie, w którym oczekuje się, że ograniczona liczba scenariuszy testowych powinna być reprezentatywna dla całej przestrzeni możliwych przypadków. Jak zapewnić tę reprezentatywność wybierając zbiór testowy? Bądź jak uzasadnić, że wybrany zbiór jest reprezentatywny? Zwłaszcza, kiedy przestrzeń wszystkich scenariuszy jest przestrzenią nieciągłą.

Jakkolwiek praca została napisana z dużą dbałością edycyjną, to pojawiło się kilka niedociągnięć.

1. Brak mi podsumowań poszczególnych rozdziałów, w których wyróżniono by istotne wnioski i fakty ułatwiające czytelnikowi śledzenie myśli autora i rozeznanie etapu procesu przeprowadzanego dowodu tezy i osiągnięcia celów wskazanych we rozdziale trzecim.

2. Niektóre skróty literowe pojawiają się przed późniejszym ich rozwinięciem, lub wyjaśnieniem. Na przykład QoS pojawia się na stronie 14, wyjaśnione jest na stronie 20.
3. Strona 19 podrozdział 2.3. Zapis w nawiasie (*SZODZ, ang. Resource-Constrained Project Scheduling Problem*) sugeruje angielskie wyjaśnienie skrótu, który jest skrótem polskim. Polski termin podany jest wcześniej, ale poszczególne słowa terminu powinny być tam zapisane poczynając od dużych liter.

#### 4. Wiedza kandydata

Podjęty przez doktoranta w rozprawie problem naukowy wymaga wiedzy i umiejętności z wielu obszarów badawczych, w szczególności z zakresu programowania rekonfigurowalnych systemów wbudowanych, szeregowania zadań, czy technik ewolucyjnych. Zawartość treści w pierwszych dwóch rozdziałach pracy zapoznającą czytelnika z aktualnym stanem wiedzy dziedzinowej, a także opis jej realizacji w rozdziałach następnych wskazuje na szeroką wiedzę autora rozprawy w powyższych obszarach i umiejętność z niej korzystanie. Ponadto doktorant wykazał się poprawnością terminologiczną i szeroką znajomością literatury przedmiotu.

#### 5. Podsumowanie

Biorąc pod uwagę opinie zaprezentowane w poprzednich punktach, w tym nieliczne uwagi krytyczne, w części natury dyskusyjnej i edytorskiej, i wymagania zdefiniowane przez artykuł 187 ustawy z dnia 20 lipca 2018r. Prawo o szkolnictwie wyższym i nauce (z późniejszymi zmianami) **stwierdzam, że:** rozprawa doktorska pana mgr inż. Leszka Ciopińskiego zawiera oryginalne rozwiązanie problemu naukowego, kandydat posiada ogólną wiedzę teoretyczną w dyscyplinie *informatyka techniczna i telekomunikacja* i posiada umiejętność samodzielnego prowadzenia pracy naukowej.

W związku z powyższym **wnoszę** o przyjęcie rozprawy przez odpowiednią Komisję powołaną przez Radę Dyscypliny Naukowej Informatyka Techniczna i Telekomunikacja Politechniki Warszawskiej i dopuszczenie jej do publicznej obrony.

