

MECHANIZM SZEREGOWANIA PROCESÓW W SYSTEMIE WIELOAGENTOWYM WSPOMAGAJĄCYM PROJEKTOWANIE UKŁADÓW STEROWANIA

Przemysław SPYCHALSKI¹, Ryszard ARENDT²

Politechnika Gdańska, Wydział Elektrotechniki i Automatyki
1. tel.: 58 347 2139, e-mail: przspsych@student.pg.gda.pl
2. tel.: 58 347 2157, e-mail: ryszard.arendt@pg.gda.pl

Streszczenie: W artykule przedstawiono mechanizm szeregowania procesów (symulacji) w systemie wieloagentowym wspomagającym projektowanie układów sterowania. Zaimplementowany algorytm ma za zadanie przydzielanie poszczególnych procesów do jednostek obliczeniowych, tak aby minimalizować czas przetwarzania zapytań przez system. Szeregowanie procesów pozwoliło na zwiększenie wydajności systemu wieloagentowego na poziomie 245%.

Słowa kluczowe: szeregowanie procesów, system wieloagentowy, wspomaganie projektowania.

1. WPROWADZENIE

Systemy wieloagentowe składają się z wielu niezależnych, autonomicznych jednostek (agentów), które współpracują ze sobą w celu realizacji określonego zadania. Interakcje pomiędzy agentami zachodzą w sposób niedeterministyczny, a dane w systemie przesyłane są asynchronicznie. Przypadkowość działania systemu wieloagentowego może w pewnych warunkach doprowadzić do sytuacji, w której dostępne zasoby sprzętowe będą wykorzystywane do rozwiązywania zadań obliczeniowych w sposób mało efektywny.

We wcześniejszych publikacjach przedstawiono system wieloagentowy składający się między innymi z agentów symulacyjnych, którym przydzielane były złożone zadania obliczeniowe [1-3]. Zaobserwowano, że metoda odpowiedzialna za listowanie dostępnych agentów platformy UMAP opiera się głównie na kolejności ich rejestracji w DF (Dictionary Facilitator). Z tego powodu kilka procesów obliczeniowych może zostać przydzielonych do pojedynczego agenta (który kilkakrotnie znalazł się pierwszy na liście), podczas gdy inni pozostają w stanie bezczynności. Sytuacja ta odbija się niekorzystnie na wydajności systemu wieloagentowego.

W artykule przedstawiono koncepcję mechanizmu szeregowania procesów (symulacji) w systemie wieloagentowym wspomagającym projektowanie układów sterowania. Algorytm ten ma na celu przydzielanie poszczególnych procesów do jednostek obliczeniowych, tak aby minimalizować czas przetwarzania zapytań przez system. W dalszej części opisano zaimplementowany mechanizm szeregowania procesów oraz dokonano testów wydajnościowych systemu wieloagentowego z aktywnym algorytmem.

2. MECHANIZM SZEREGOWANIA PROCESÓW

Szeregowanie procesów (alternatywnie nazywane planowaniem, ang. Scheduling) jest to strategia udostępniania zasobów, takich jak czas procesora, obszar pamięci operacyjnej dla określonego procesu. Znanych jest wiele algorytmów, które odpowiadają za schemat przypisywania procesów do zasobów sprzętowych. Najprostszym z nich jest FCFS (ang. First-Come, First-Served), który polega na wykonywaniu procesów w takiej kolejności jak zostały przysłane do jednostki obliczeniowej. Kolejnym mechanizmem jest SJN (ang. Shortest-Job-Next), czyli uruchamianie w pierwszej kolejności procesów, które mają najkrótszy czas wykonywania. Ostatnim z podstawowych algorytmów szeregowania jest planowanie rotacyjne (ang. Round Robin). W przeciwieństwie do wcześniej przedstawionych technik używany jest tutaj mechanizm wywłaszczania, w wyniku czego każdy proces cyklicznie otrzymuje określony kwant czasu dostępu do procesora, co z kolei pozwala na quasi-równoległe przetwarzanie wszystkich procesów [4].

Szeregowanie procesów w systemach wieloagentowych jest bardziej skomplikowane ze względu na rozproszenie systemu oraz liczbę jednostek biorących udział w procesie planowania. Wyróżnia się trzy podstawowe kryteria podziału tego zagadnienia, biorąc pod uwagę samodzielność, komunikację i kooperację agentów w procesie szeregowania zadań [5]. Aby uprościć schemat przydzielania zadań agentom symulacyjnym zastosowano scentralizowane podejście szeregowania procesów. Za wykonywanie zaimplementowanego algorytmu odpowiedzialny jest agent Master, który pośredniczy pomiędzy agentami interfejsu i agentami obliczeniowymi. W momencie kiedy od warstwy interfejsu odbierane są kolejne żądania wymagające obliczeń symulacyjnych, wówczas Master szereguje je i przydziela w taki sposób, aby czas przetworzenia wszystkich zapytań był jak najkrótszy. Zostało to osiągnięte poprzez odpytywanie agentów symulacyjnych o poziom wykorzystania procesora w danej chwili i przydzielenie zadania jednostce najmniej obciążonej. Podstawowy opis działania algorytmu:

- 1) Odbierz żądanie od agenta interfejsu.
- 2) Dodaj żądanie na koniec kolejki oczekujących.
- 3) Odpytaj agentów symulacyjnych o procentowy poziom wykorzystania procesora w danej chwili.

```
c:\> wmic cpu get loadpercentage
```

- 4) Jeżeli najmniejszy procentowy wskaźnik wykorzystania CPU wszystkich jednostek jest większy niż 90% (brak dostępnych zasobów), wówczas odczekaj 5 sekund i idź do punktu 3.
- 5) Wyślij żądanie znajdujące się na początku kolejki do najmniej obciążonej jednostki (metoda FIFO), następnie idź do punktu 1.

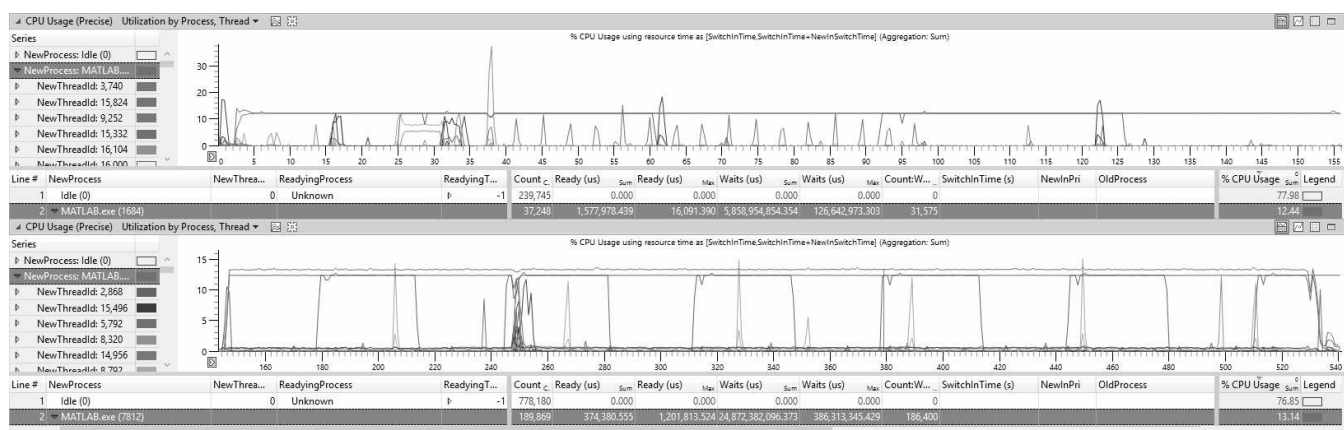
3. TESTY WYDAJNOŚCIOWE

W celu oceny jakości opracowanego rozwiązania przeprowadzone zostały testy wydajnościowe systemu wieloagentowego. Metodologia pomiarów zakładała wysłanie trzech jednakowych żądań obliczeń symulacyjnych z warstwy interfejsu i zmierzenie czasu otrzymania odpowiedzi od warstwy obliczeniowej (składającej się z trzech agentów). Monitorowano przy tym stan wykorzystania procesora pierwszej jednostki, na której uruchomiony został agent symulacyjny z zastosowaniem narzędzia Windows Performance Recorder (Rys. 1.). Procedurę pomiarową wykonano dla scenariusza z aktywnym (górnym wykres) oraz nieaktywnym (dolnym wykres) mechanizmem szeregowania procesów. Zaobserwowano wzrost wydajności systemu wieloagentowego po uruchomieniu algorytmu, wyrażony zredukowaniem czasu przetwarzania żądań z 542 do 155

sekund. Spowodowane było to uszeregowaniem procesów tak, że na każdego agenta przypadł tylko jeden proces do przetworzenia (kiedy algorytm był nieaktywny wszystkie trzy procesy zostały przydzielone pojedynczemu agentowi). Specyfikacja sprzętowa agentów symulacyjnych: Intel Core i7 vPro 2.7 GHz, DDR 16GB, Intel SSDSCKJF180A5.

4. PODSUMOWANIE

W artykule przedstawiono mechanizm szeregowania procesów (symulacji) w systemie wieloagentowym wspomagającym projektowanie układów sterowania. Mechanizm ten oparty został na założeniu, że przychodzące żądania z warstwy interfejsu powinny zostać uszeregowane i wykonane przez najmniej obciążoną jednostkę (pod względem wykorzystania CPU) w warstwie obliczeniowej. Jeżeli natomiast wszystkie jednostki są zajęte, wówczas żądania powinny zostać dodane do kolejki i czekać na zwolnienie zasobów. Opracowany algorytm stanowi łatwą w implementacji alternatywę do powszechnie znanych algorytmów szeregowania. Zastosowanie mechanizmu szeregowania procesów przyczyniło się do zwiększenia wydajności systemu wieloagentowego na poziomie 245% poprzez skrócenie czasu wykonywania procesów (Rys. 1.).



Rys. 1. Porównanie wydajności systemu wieloagentowego z aktywnym i nieaktywnym algorytmem szeregowania procesów

5. BIBLIOGRAFIA

1. Arendt R., Kopczyński A., Spychalski P.: Centralized and Distributed Structures of Intelligent Systems for Aided Design of Ship Automation, Proceedings of 38th International Conference on Information Systems Architecture and Technology 2017
2. Arendt R., Spychalski P.: An application of multi-agent system for ship's power systems design, Proceedings of 20th International Conference Transport Means 2016
3. Spychalski, P., Arendt R.: Rozproszony system wieloagentowy wspomagający projektowanie wybranych układów sterowania, Zeszyty Naukowe Wydziału Elektrotechniki i Automatyki Politechniki Gdańskiej 2016
4. Goel N., Garg R.: A Comparative Study of CPU Scheduling Algorithms, International Journal of Graphics & Image Processing 2012
5. Weerd M., Clement B.: Introduction to planning in multiagent systems, Multiagent and Grid Systems 2009

MECHANISM OF PROCESSES SCHEDULING IN MULTI-AGENT SYSTEM FOR AIDED DESIGN OF CONTROL SYSTEMS

In this paper the new mechanism of processes scheduling in multi-agent systems is presented. This mechanism is based on the assumption that incoming requests shall be scheduled and processed by the least loaded unit. Implementation of the algorithm improved the multi-agent system performance at the level of 245% by reducing the time of request processing.

Keywords: process scheduling, multi-agent system, aided design.