

Barbara Zakrzewska¹, Piotr Baniukiewicz², Zdzisław Jaworski¹

e-mail: zakrzewska@zut.edu.pl

¹Institut Inżynierii Chemicznej i Procesów Ochrony Środowiska, Wydział Technologii i Inżynierii Chemicznej, Zachodniopomorski Uniwersytet Technologiczny, Szczecin²Katedra Elektrotechniki Teoretycznej i Informatyki, Wydział Elektryczny, Zachodniopomorski Uniwersytet Technologiczny, Szczecin

Modelowanie wieloskalowe systemu procesowego. Numeryczne badania wstępne

Wstęp

Coraz częściej w modelowaniu systemu procesowego stosuje się podejście wieloskalowe, co jest związane z wymaganiami klienta dotyczącymi produktu o określonych własnościach. Projektant znając rynkowe zapotrzebowania na cechy produktu, skupia się na metodologii i narzędziach prowadzących do produktu o żądanych cechach. Postępowanie takie może być przeprowadzone na różnych poziomach szczegółowości, często po przeprowadzeniu analizy procesu w różnych skalach długości i czasu [1–3].

W pracy wykorzystano dwa różne narzędzia numeryczne: wspomagana komputerowo inżynieria systemu procesowego (*Process System Engineering*, PSE) oraz numeryczną dynamikę płynów (*Computational Fluid Dynamics*, CFD). Tradycyjnie pakiety CFD i PSE wykorzystywane są odrębnie. Jednakże integracja tych dwóch metod numerycznych może przynosić wymierne korzyści. Pakiety programów do symulacji procesów umożliwiają zarówno symulację procesów jednostkowych, jak również całych linii technologicznych. Ostatnia generacja programów jest bardzo zaawansowana i pozwala na symulację skomplikowanych procesów, gdzie występuje wiele składników, wiele faz i zachodzą reakcje chemiczne. W tej metodzie modelowania stosowane są zazwyczaj duże uproszczenia oraz zakładane są wielkości procesowe. Pakiety programów CFD wykorzystywane są do modelowania pojedynczych urządzeń, lub ich fragmentów wówczas, gdy wymagane są szczegółowe wyniki modelowania, czyli najczęściej wtedy, gdy wydajność urządzenia jest uzależniona od dynamiki płynów. Nie opracowano jeszcze ogólnie uznanych kryteriów i narzędzi łącznego wykorzystania tych dwóch metod numerycznych. Ogólna koncepcja integracji kodów CFD oraz PSE, wykorzystywanej podczas modelowania wieloskalowego jest taka, że system procesowy modelowany jest za pomocą programów do symulacji procesów (np. *gPROMS*, *Aspen Plus*, *Aspen HYSYS*), natomiast kluczowy aparat – za pomocą programów CFD (np. *Fluent*, *CFX*, *Comsol*) [4, 5]. Przykłady aplikacyjne integracji tych dwóch różnych metod modelowania zostały szeroko omówione w pracy [6].

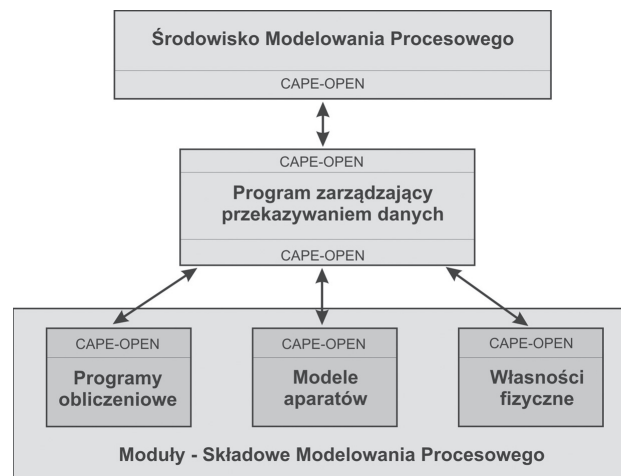
Celem niniejszej pracy było stworzenie i sprawdzenie poprawności działania nakładki programowej (interfejsu) pozwalającej na poprawne przekazywanie danych pomiędzy dwoma różnymi kodami numerycznymi stosowanymi w różnych skalach długości i czasu. Każda z tych metod dostarcza innego rodzaju dane procesowe. Można się spodziewać, że ten sposób modelowania, który będzie dalej rozwijany, przyczyni się do optymalizacji działającego systemu technologicznego.

Standardy CAPE-OPEN

W niniejszej pracy do integracji kodów CFD i PSE wykorzystano standardy CAPE-OPEN. Powstały one w celu ujednoczenia połączeń między różnymi programami wykorzystywanymi w szeroko pojętej wspomaganej komputerowo inżynierii procesowej oraz dla bardziej efektywnej integracji różnych metod modelowania. W chwili obecnej są one rozwijane i propagowane przez konsorcjum CAPE-OPEN *Laboratories Network* (CO-Lan) (www.colan.org). Standard CAPE-OPEN definiuje zasady i interfejsy do integracji i współdziałania programów. Zaletami jego stosowania są: i) łatwe połączenie, typu *plug and play* pomiędzy różnymi programami, niezależnie od ich architektury i języka programowania, ii) możliwość użycia interfejsów jako części wspólnej

struktury aplikacyjnej, co pozwala na wymianę informacji procesowej wewnątrz domeny aplikacyjnej [7].

Według standardów CAPE-OPEN programy, pomiędzy którymi wymieniane są dane, są dzielone na dwie grupy. Pierwsza to środowisko modelowania procesowego reprezentowane przez nadrzędny symulator i tu najczęściej występuje program do symulacji procesów, natomiast druga złożona jest z modułów – składowych modelowania procesowego [8], na przykład odrębnych programów obliczeniowych, modeli matematycznych aparatów, czy programów do obliczeń własności termodynamicznych. Do tej grupy można również zaliczyć kody CFD, za pomocą których konstruowany jest szczegółowy model danego aparatu. W chwili obecnej niewiele jest przykładów takiej integracji. Zazwyczaj są one związane z *Advanced Process Engineering Co-Simulator* (APECS) [9]. Schematyczny, bardzo ogólny sposób połączenia między programami pokazano na rys. 1. Komunikacja pomiędzy środowiskiem modelowania procesowego a poszczególnymi modułami odbywa się za pomocą *middlewares*, takich jak CORBA (*Common Object Request Broker Architecture*) firmy *Object Management Group* (OMG) oraz COM (*Component Object Model interface definition language*) i *.Net*, oba firmy *Microsoft*. Wymienione narzędzia programistyczne przedstawiono w pracy [10].



Rys. 1. Integracja oprogramowania w oparciu o CAPE-OPEN

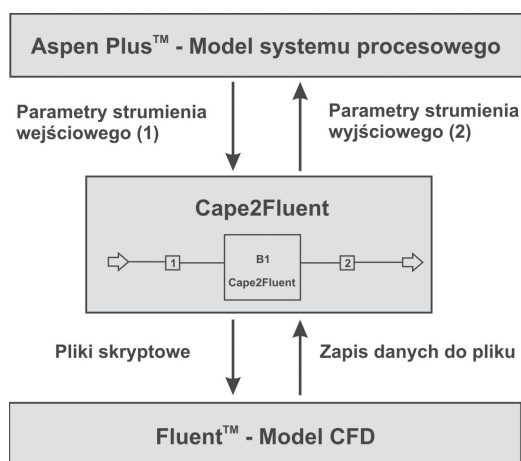
Koncepcja i aplikacja interfejsu *Cape2Fluent*

Niniejsza praca związana jest ściśle z modelowaniem wieloskalowym w odniesieniu do systemu technologicznego do produkcji nawozów sztucznych, gdzie kluczowym elementem był reaktor rurowy. Praca ta stanowi pierwszy etap badań nad integracją dwóch różnych narzędzi numerycznych wykorzystywanych do symulacji procesów w stanie ustalonym (*Aspen Plus*) oraz numerycznej dynamiki płynów (*Fluent*). W wyniku badań został stworzony otwarty interfejs komunikacji międzyprogramowej o nazwie *Cape2Fluent* z wykorzystaniem CAPE-OPEN. W obecnej wersji program *Aspen Plus* współpracuje z innymi programami w oparciu o standardy CAPE-OPEN, natomiast program *Fluent* nie wspiera tego standardu. Był to główny problem, który należało przezwyciężyć. W tym przypadku zaistniała potrzeba

znalezienia innego sposobu przesyłania i odbierania danych z *Fluenta*. Powstały program jest więc nie tylko interfejsem pomiędzy tymi dwoma narzędziami numerycznymi, ale także interfejsem pomiędzy samym standardem CAPE-OPEN, a innymi metodami importu i eksportu danych, które nie są wspierane przez ten standard.

Zgodnie z ideą CAPE-OPEN środowiskiem modelowania procesowego był program *Aspen Plus*, natomiast model aparatu uzyskany za pomocą programu *Fluent* był jednym z bloków wstawianych do modelu systemu procesowego. Pierwszym krokiem przy jego włączaniu za pomocą interfejsu *Cape2Fluent* było załadowanie w programie *Aspen Plus* biblioteki modeli zgodnych z CAPE-OPEN. Wówczas to w grupie wszystkich dostępnych modeli aparatów (bloków) znalazł się *Cape2Fluent*. Kolejnym krokiem było zdefiniowanie systemu aparatów wraz z powiązaniem z nimi strumieniami wejściowymi i wyjściowymi (*flowsheet*). W miejsce aparatu, którego model był stworzony w programie *Fluent*, do systemu wprowadzano *Cape2Fluent*. Na obecnym etapie badań interfejs pozwalał na wstawienie do systemu bloku o jednym wejściu i jednym wyjściu. Należy dodać, że niezależnie od opisanych czynności model szczegółowy aparatu w programie *Fluent* tworzony był oddzielnie.

Interfejs *Cape2Fluent* pozwala na wykonanie obliczeń w sposób sekwencyjny, czyli tak, jak jest to prowadzone w programie *Aspen Plus* w stanie ustalonym. W testowanym przypadku stworzono *flowsheet* składający się z jednego bloku, którym był *Cape2Fluent* z doprowadzonymi do niego strumieniami: wejściowym i wyjściowym. Schemat przekazywania danych podczas wykonywanych obliczeń przedstawiono ogólnie na rys. 2. Z programu *Aspen Plus* wprowadzono do *Cape2Fluent* dane strumienia wejściowego (1) takie jak np. strumień masowy, jego temperatura, ciśnienie i skład. Następnie interfejs generował plik skryptowy, w którym zapisane były operacje takie jak: definicja warunków brzegowych, co służy przekazaniu wartości parametrów strumienia z programu *Aspen Plus* do modelu w programie *Fluent*, polecenia: wykonania symulacji, zapisania określonych wyników na dysk oraz zamknięcia programu. Kolejny interfejs *Cape2Fluent* uruchamiał *Fluenta*, wczytywał do niego plik skryptowy i odbierał wyniki, które dekodował i przekazywał obliczone parametry z powrotem do programu *Aspen Plus*, jako parametry strumienia wyjściowego (2).



Rys. 2. Schemat przekazywania danych z wykorzystaniem interfejsu *Cape2Fluent*

Testowanie poprawności działania stworzonego interfejsu i przekazywania danych pomiędzy programami *Aspen Plus* i *Fluent* przeprowadzono na prostym systemie. Złożony on był z dwóch strumieni: wejściowego i wyjściowego oraz jednego bloku, którym był *Cape2Fluent*. Z kolei szczegółowym modelem CFD był model rury o rzadkiej siatce numerycznej, zbudowanej z 1500 komórek obliczeniowych. Jako waru-

nek graniczny na wlocie zdefiniowano strumień masowy czynnika wlotowego i jego temperaturę. Modelowano przepływ laminarny wody.

Z programu *Aspen Plus* do *Fluenta* przekazywane były wartości strumienia masowego i temperatury czynnika na wlocie do rury reaktora. Po wykonanych obliczeniach w programie *Fluent* przekazywane były zwrótnie do *Aspena* wartości prędkości oraz temperatura czynnika na wlocie z rury reaktora. Wartości ciśnienia i udziału objętościowego strumienia wejściowego nie były wprowadzane do modelu CFD i dlatego zostały przepisane ich niezmiennione wartości do parametrów strumienia wyjściowego z bloku.

W wyniku zastosowania programu zarządzającego przekazywaniem danych *Cape2Fluent* uzyskano te same wartości parametrów strumienia wejściowego i wyjściowego, wyświetlone jako wyniki symulacji całego systemu w programie *Aspen Plus*.

Kolejnym krokiem będzie sprawdzenie poprawności działania interfejsu *Cape2Fluent*, gdy do systemu procesowego zostanie włączony model CFD reaktora z krzyżowym wlotem składników [11].

Posumowanie i wnioski

Podczas badań wstępnych nad integracją dwóch różnych narzędzi modelowania: programów *Aspen Plus* i *Fluent* stworzono nakładkę programową (interfejs) *Cape2Fluent*. Pozwala ona na przekazywanie danych pomiędzy tymi programami z wykorzystaniem standardów CAPE-OPEN, jak również pomiędzy samym standardem CAPE-OPEN, a innymi metodami importu i eksportu danych, które nie są przez niego wspierane. Skuteczność tego rozwiązania oraz jego uniwersalność została potwierdzona nie tylko poprzez współpracę z programem *Aspen Plus*, ale także z innymi programami do symulacji procesów chemicznych.

Przeprowadzone testy dowiodły, że działanie stworzonego interfejsu jest poprawne. Zatem numeryczne badania wstępne w kierunku integracji oprogramowania stosowanego w różnych skalach długości i czasu można uznać za zakończone sukcesem. Jest to pierwszy krok do stworzenia wieloskalowego systemu procesowego. Planowane są dalsze badania aplikacyjne i udoskonalenia interfejsu.

LITERATURA

- [1] D. G. Vlachos, A. B. Mhadeshwar, N. S. Kaisare: *Comp. Chem. Eng.*, 30, 1712 (2006).
- [2] J.-C. Charpentier: *Comp. Chem. Eng.*, 33, 936 (2009).
- [3] M. Fermeiglia, S. Priol: *Comp. Chem. Eng.*, 33, 1701 (2009).
- [4] F. Bezzo, S. Macchietto, C. C. Pantelides: *Comp. Chem. Eng.*, 24, 653 (2000).
- [5] S. E. Zitney, M. Syamlal: *Integrated process simulation and CFD for improved process engineering*, in: Grievink J. and van Schijndel J. (Eds.), 12th ESCAPE, *Computer Aided Chemical Engineering*, 10, 397 (2002).
- [6] Z. Jaworski, B. Zakrzewska: *Chem. Proc. Eng.*, 29, 567 (2008).
- [7] W. M. Barrett, J. Yang: *Comp. Chem. Eng.*, 30, 191 (2005).
- [8] R. Morales-Rodriguez, R. Gani, S. Déchelotte, A. Vacher, O. Baudouin: *Chem. Eng. Res. Des.*, 86, 823 (2008).
- [9] D. A. Swensen, S. E. Zitney, M. Bockelie: *Development of Cape-Open unit operations for advanced power systems modeling*, AICHE 2007 Annual Meeting, November 4th-9th, 2007, Salt Lake City, Utah, session TD001, prezentacja dostępna na (10.04.2010): http://www.colan.org/Conferences/Y07_4thCOUS_398e.pdf.
- [10] E. Rój: *Przem. Chem.*, 82, 1253 (2003).
- [11] Ł. Gralla, Z. Jaworski, B. Zakrzewska, M. Klejny: *Inż. Ap. Chem.*, 48, nr 6, 74 (2009).

Praca naukowa finansowana ze środków na naukę w latach 2007–2010, wykonana w ramach projektu badawczego rozwojowego nr R14 007 03.