



ANDRZEJ PASZKIEWICZ¹, JAN WĘGRZYN²,
ROBERT PEKALA³

Możliwości zastosowania oprogramowania Scilab do projektowania i modelowania sieci komputerowych

Possibilities of Using Scilab Software for Designing and Modeling Computer Networks

¹ ORCID: 0000-0001-7573-3856, doktor inżynier, Politechnika Rzeszowska im. Ignacego Łukasiewicza, Wydział Elektrotechniki i Informatyki, Zakład Systemów Złożonych, Polska

² Student, Politechnika Rzeszowska, Polska

³ ORCID: 0000-0003-0530-0005, doktor, Państwowa Wyższa Szkoła Techniczno-Ekonomiczna w Jarosławiu, Instytut Inżynierii Technicznej, Polska

Streszczenie

W artykule zaprezentowano możliwości wykorzystania środowiska *open source* Scilab do projektowania oraz modelowania sieci komputerowych. Środowisko to może być alternatywą dla komercyjnych rozwiązań w zakresie symulacji protokołów i algorytmów wykorzystywanych w sieciach komputerowych.

Słowa kluczowe: Scilab, sieci komputerowe, projektowanie, modelowanie, *open source*

Abstract

The article presents the possibilities of using the Scilab open source environment for designing and modeling computer networks. This environment can be an alternative to commercial solutions in the field of simulation of protocols and algorithms used in computer networks.

Keywords: Scilab, computer networks, designing, modeling, open source

Wstęp

Obecnie jesteśmy świadkami gwałtownego rozwoju sieci komputerowych. Pojawiają się nowe technologie, zatwierdzane są nowe standardy komunikacyjne, a producenci urządzeń sieciowych dostarczają rozwiązania charakteryzujące się przepustowością na poziomie kilkudziesięciu Tb/s. Równocześnie wzrasta liczba urządzeń końcowych. Przewiduje się, że w 2022 r. do globalnej sieci internet podłączonych będzie ponad 50 mld urządzeń, z czego istotną część stanowią będą

rozwiązania mobilne. Wynika to z faktu m.in. upowszechniania się rozwiązań Internetu Rzeczy (*Internet of Thing* – IoT). Wraz z przyrostem ilościowym urządzeń rośnie także zapotrzebowanie na przepustowość łączy i węzłów sieciowych. Szacunkowe dane wskazują, że globalny ruch IP zwiększy się trzykrotnie w ciągu najbliższych 5 lat. Taki stan rzeczy powoduje, że potrzebne są adekwatne metody i środki związane z projektowaniem sieci komputerowych oraz modelowaniem procesów w nich zachodzących. Biorąc pod uwagę, że podstawy projektowania sieci komputerowych wywodzą się z teorii grafów, a zagadnienia związane z ruchem sieciowym odnoszą się do teorii przepływów, niezbędne do tego celu wydają się narzędzia matematyczne. W tym zakresie dostępne są rozwiązania komercyjne oraz dystrybuowane na zasadzie licencji *open source*. W artykule przedstawione zostanie darmowe oprogramowanie Scilab, którego potencjał wskazuje na szerokie zastosowanie w nauce, inżynierii oraz dydaktyce. Uwzględniając fakt, iż w wielu przypadkach wykorzystanie tego typu narzędzi wiąże się z wysokimi opłatami licencyjnymi, zwłaszcza w obszarze edukacji, ograniczenie kosztów wdrożenia nowego rozwiązania ma istotne znaczenie. Niemniej najważniejszym czynnikiem decydującym o ewentualnym zastosowaniu tego oprogramowania jest dostępny zakres funkcjonalny oraz możliwości jego rozwoju. W artykule zaprezentowano możliwości oraz potencjalny zakres zastosowania środowiska Scilab w zagadnieniach projektowania oraz modelowania sieci komputerowych.

Charakterystyka środowiska Scilab

Oprogramowanie Scilab dystrybuowane jest na licencji *open source* z przeznaczeniem w głównej mierze dla inżynierów i naukowców. Jest ono wykorzystywane do obliczeń numerycznych. Posiada swój własny język programowania wysokiego poziomu, a także interpreter. Dostępne jest na platformy GNU/Linux, Mac OS X i Windows XP/Vista/7/8/10. Biorąc pod uwagę zakres funkcjonalny, jego możliwości są znacznie mniejsze niż porównywalnych komercyjnych aplikacji (np. Sandeep, 2017). Należy jednak zadać pytanie, czy w zakresie np. dydaktycznym dostępny potencjał nie jest wystarczający. Odpowiedź na nie zostanie udzielona na końcu tego artykułu.

Scilab posiada wiele wbudowanych funkcji i narzędzi do obliczeń numerycznych, przeprowadzania symulacji czy też wizualizacji wyników, w tym możliwe są do wykonania operacje bitowe, na macierzach, na liczbach zmiennoprzecinkowych, a także operacje z zakresu matematyki dyskretnej, statystyki, konwersji liczb, przetwarzania sygnałów, logarytmów i funkcji wykładniczych oraz wiele innych. Jedną z cech Scilab jest jego interoperacyjność, tzn. możliwość wykorzystania innych aplikacji za pomocą rozszerzeń z użyciem języków: Java, Python, Tcl Tk, Fortran, C i C++. Zwiększa to potencjał wykorzystania Scilab do celów dydaktycznych i naukowych poprzez integrację z większym środowiskiem aplikacyjnym, w którym następować będzie wewnętrzna wymiana danych pomiędzy poszczególnymi modułami.

Scilab zapewnia możliwość zarządzania modułami wyposażonymi w specjalistyczne dodatki i gotowe funkcje. Jest to realizowane za pomocą ATOMS (*AuTomatic mOdules Management for Scilab*). W ramach tego repozytorium zawartych jest wiele modułów rozszerzeń. Jednak z punktu widzenia projektowania i modelowania sieci komputerowych interesujące są trzy z nich, tj.:

1. **Metanet** jest narzędziem Scilab do obliczeń na grafach i sieciach. Zostały tutaj zaimplementowane algorytmy do rozwiązywania klasycznych problemów grafów i minimalnego kosztu przepływu.

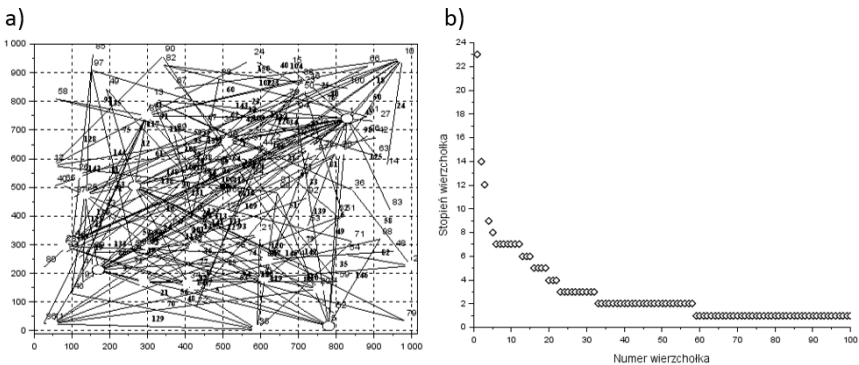
2. **Network Topology Generator (NTG)** to moduł dedykowany analizie topologii sieciowej. Pozwala na generowanie losowych topologii, wykorzystując przy tym algorytmy: Locallity, Waxman, Barabasi-Albert oraz modelu hierarchicznego. Celem tego modułu są badania wpływu algorytmów routingu na wydajność protokołów transmisji danych używanych w sieci. Posiada wbudowane algorytmy do wyliczania najkrótszych ścieżek pomiędzy sieciami, takie jak Dijkstry, Bellmana-Forda, Flood, BFS, Prima i Floyd-Warshalla.

3. **Network Analysis and Routing eVALuation (NARVAL)** jest rozbudowaną wersją NTG. Posiada wiele dodatkowych funkcji dotyczących operacji na grafach i macierzach oraz badania ich parametrów. Został wzbogacony o dodatkowe protokoły routingu, jak Ad hoc On-Demand Vector Routing, ARC czy RPL. Udostępnione zostały także funkcje związane z szyfrowaniem oraz wizualizacją grafów. Najważniejsza dodana grupa procedur dotyczy symulowania transmisji danych pomiędzy dwiema sieciami.

Wykorzystanie Scilab do projektowania oraz modelowania sieci komputerowych

W rozdziale tym przedstawiono przykłady zastosowania środowiska Scilab w obszarach projektowania sieci komputerowych. Pierwszy taki zasadniczy obszar to tworzenie oraz analiza topologii sieciowych. Omawiane narzędzie doskonale nadaje się do tworzenia struktur topologicznych odwzorowujących środowisko fizyczne sieci komputerowych, jak również strukturę logiczną związaną z komunikacją zachodzącą w nich. Można tworzyć topologie klasyczne, np. krata, toroid, drzewo, drzewo binarne, gwiazda, hipersześcian, pierścienie itp., ale również można generować losowe struktury w oparciu o jedną z dostępnych procedur, np. Barabasi-Albert. Bazuje ona na jednym z modeli wywodzących się z teorii systemów złożonych (Nicolis, 2012; Albert, 2002) i generuje strukturę topologiczną na bazie grafu losowego. Główną zasadą tej procedury jest założenie, że im wyższy stopień danego węzła, tym większe prawdopodobieństwo podłączenia do niego kolejnych węzłów. Mechanizm ten związany jest z tzw. preferencyjnym dołączaniem. Na rysunku 1a przedstawiono przykładową topologię stworzoną w oparciu o ten algorytm na bazie 100 wierzchołków, wykorzystując funkcję *Barabasi-Albert* z modułu NARVAL. Na rysunku 1b pokazano rozkład stopni poszczególnych wierzchołków tej topologii, który odpowiada

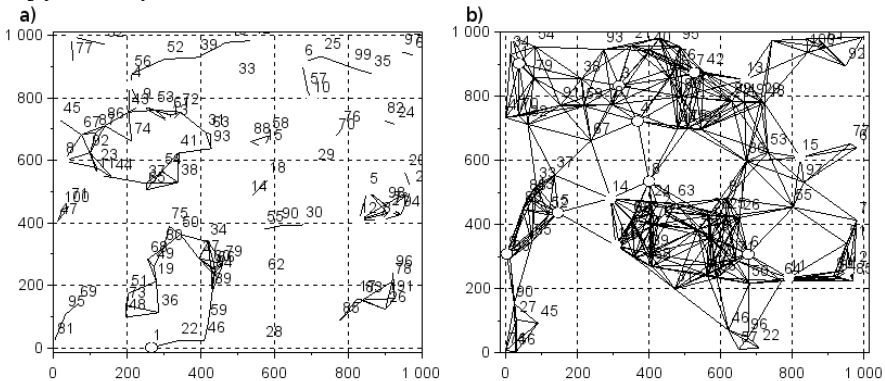
rozkładowi zgodnemu z prawem potęgowym (Albert, 1999). Inną procedurą może być *Locality* – tworzy ona losową topologię z zadaną liczbą węzłów rozmieszczonych na danym obszarze. Połączenia pomiędzy poszczególnymi węzłami powstają, gdy węzły te znajdują się w swoim zasięgu, tzn. w ustalonej odległości zgodnie z przyjętą wartością parametru promienia R . Podczas generowania topologii na bazie tej metody można uzyskać strukturę niespójną w przypadku, gdy przyjęty promień będzie zbyt mały, tak jak pokazano to na rysunku 2a. Mechanizm ten nadaje się do projektowania sieci bezprzewodowych oraz infrastruktury IoT.



Rysunek 1. Struktura sieciowa o 100 wierzchołkach na bazie algorytmu *Barabasi-Albert* (a) oraz rozkład stopni wierzchołków dla wygenerowanej topologii zgodny z prawem potęgowym (b)

Źródło: opracowanie własne.

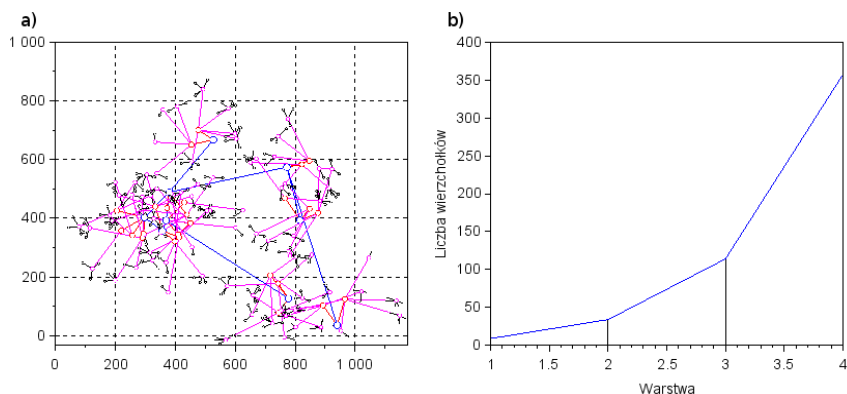
Na rysunku 2 przedstawiono dwie wygenerowane topologie wykorzystujące funkcję *Locality* z modułu NARVAL.



Rysunek 2. Struktura sieciowa o 100 wierzchołkach i promieniu 100 (a) oraz struktura sieciowa o 100 wierzchołkach i promieniu 200 (b)

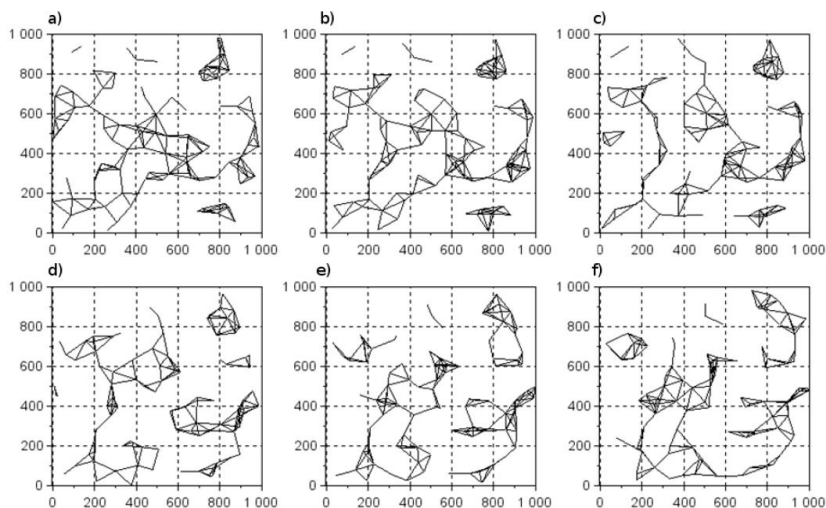
Źródło: opracowanie własne.

Kolejna procedura – Waxman – to model, który w najlepszy sposób odzwierciedla realną strukturę sieciową. Bierze on pod uwagę takie parametry, jak: odległość euklidesowa oraz maksymalna dopuszczalna odległość między dwoma wierzchołkami. Na podstawie danych wejściowych wylicza prawdopodobieństwo istnienia połączeń w danej topologii. Istnieją różne jego modyfikacje i odmiany, m.in. pozwalające na tworzenie struktur hierarchicznych. Na rysunku 3 przedstawiono topologię wygenerowaną funkcją *Waxman Connex H* z modułu NARVAL.



Rysunek 3. Wygenerowana struktura sieciowa o 357 wierzchołkach i 4 warstwach (a) oraz liczba węzłów przypadająca na każdą warstwę w topologii (b)

Źródło: opracowanie własne.



Rysunek 4. Symulacja mobilności węzłów na bazie funkcji GraphBS z przyjętym zasięgiem komunikacji o wartości 120. Kolejne kroki symulacji przebiegają w kolejności od a) do f)

Źródło: opracowanie własne.

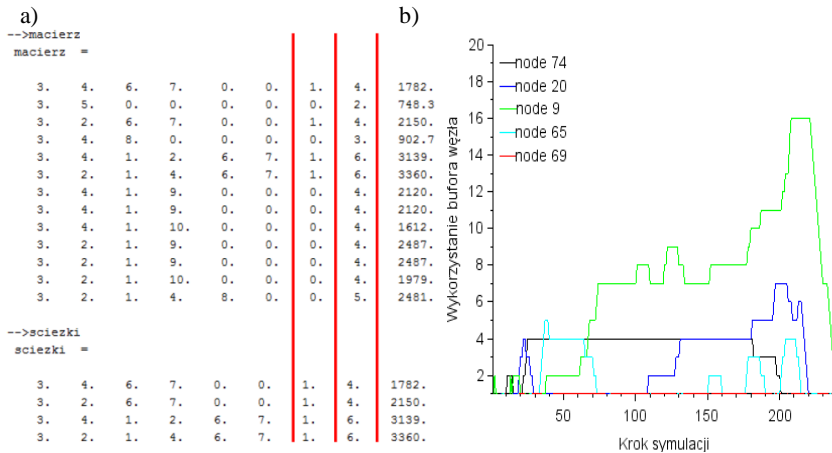
W związku z silnie rozwijanymi w obecnym czasie technologiami bezprzewodowymi obszar badań związanych z mobilnością stanowi bardzo ważny aspekt prac naukowych i dydaktycznych. W tym kontekście Scilab umożliwia symulowanie mobilności w strukturze sieciowej na bazie dynamicznie przemierzających się wierzchołków w grafie.

Na rysunku 4 pokazano przykładową sekwencję zmiany struktury topologicznej w czasie uzyskaną dzięki zastosowaniu jednej z dostępnych funkcji *GraphBS*. Stan połączeń zmienia się w czasie wraz z kolejnymi krokami algorytmu. Głównym parametrem, od którego zależy jest kreowanie bądź usuwanie krawędzi, jest zasięg komunikacji.

Przedstawione powyżej przykłady stanowią niewielki fragment możliwości wykorzystania środowiska Scilab do generowania topologii sieciowych. Z punktu widzenia projektowania i modelowania sieci komputerowych bardzo ważny aspekt dotyczy wyznaczania logicznych ścieżek komunikacji, w tym generowania struktur routingu. Scilab zapewnia możliwość testowania tego typu mechanizmów poprzez szeroką gamę zaimplementowanych algorytmów, takich jak np. Dijkstry, Bellmana-Forda, Flood, Prima, Ad hoc On-Demand Distance Vector (AODV), Breadth First Search (BFS) czy Depth First Search (DFS). Przykładowo algorytm Dijkstry jest jednym z bardziej znanych i powszechnie wykorzystywanych m.in. w protokole routingu OSPF. Dzięki tej funkcjonalności można również generować tablice routingu. Dzięki temu możliwe jest śledzenie tras pomiędzy poszczególnymi węzłami. Innym przykładem mechanizmu trasowania ruchu jest AODV stosowany głównie w sieciach bezprzewodowych oraz w IoT. Zapewnia on ścieżkę komunikacji pomiędzy oddalonymi węzłami i zestawia najkrótszą drogę do miejsca docelowego, unikając pętli topologicznych oraz reagując dynamicznie na ewentualne zmiany topologiczne. Te zmiany można śledzić w czasie rzeczywistym. Na rysunku 5a zaprezentowano przykładowe działanie funkcji reprezentującej wyszukiwanie ścieżki pomiędzy danym źródłem (węzeł 3) a miejscem docelowym (węzeł 7) przy założeniu określonego TTL. Zmienna *macierz* odzwierciedla wszystkie pakiety RREQ (*route request*) wysłane w celu wyznaczenia poszukiwanej trasy, natomiast *ścieżki* to macierz zawierająca wszystkie trasy pomiędzy dwoma wybranymi wierzchołkami, których odległość nie przekracza danego TTL. Format zapisu jest następujący: ścieżka | informacja o ścieżce | TTL | długość ścieżki. Jeżeli połączenie zostało znalezione, to parametr drugi będzie wynosić 1.

Kolejnym ciekawym obszarem funkcjonalnym, którego istotny potencjał obejmuje projektowanie i modelowanie sieci komputerowych oraz mechanizmów sieciowych, jest możliwość symulowania ruchu sieciowego, a tym samym badanie procesów zachodzących w dynamicznej strukturze sieciowej. W tym zakresie środowisko Scilab umożliwia badanie przeciążenia węzłów i ich wpływu na funkcjonowanie sieci, a także generowanie i propagowanie określonego

ruchu pomiędzy węzłami sieciowymi. Na rysunku 5b zaprezentowano – w kolejnych krokach symulacji – przykładowe obciążenie węzłów podczas transmisji pakietów TCP pomiędzy dwiema sieciami. Przedstawione powyżej przykłady zastosowań gotowych funkcji i procedur w zakresie projektowania sieci komputerowych nie obejmują ze zrozumiałych względów całego spektrum możliwości związanych z tworzeniem własnych funkcji i algorytmów dotyczących tych zagadnień.



Rysunek 5. Wyniki zapytania dla jednej z testowanych topologii dla funkcji AODV z parametrem TTL = 6 – (a) oraz wykorzystanie bufora pamięci (20 jednostek) w trakcie transmisji danych pomiędzy węzłami 69 i 74 badanej topologii (b)

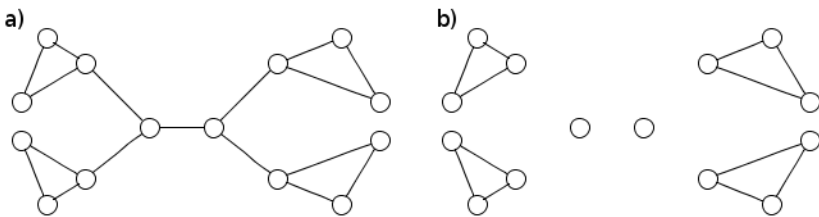
Źródło: opracowanie własne.

W związku z tym należy zaznaczyć, że dostępne środowisko umożliwiło opracowywanie oraz implementację specjalistycznych rozwiązań w zakresie m.in.:

- tworzenia topologii fizycznych i logicznych,
- analizy mechanizmów tworzenia klastrów, gridów i środowisk chmurowych,
- modelowania ruchu sieciowego i badania jego wpływu na funkcjonowanie sieci komputerowych,
- tworzenia struktur sieci społecznościowych oraz badania procesów w nich zachodzących,
- symulowania anomalii sieciowych i analizy skutków ich występowania w infrastrukturze sieciowej,
- implementacji modeli sieci bezprzewodowych,
- badania procesów związanych z interakcją rozproszonych komponentów w sieciach IoT,

– porównywania sprawności algorytmów routingu oraz drzew rozpinających dla różnych bazowych topologii.

Jednym z wymienionych powyżej obszarów zastosowań środowiska Scilab jest ogólnie pojęta analiza oraz modelowanie samych struktur sieci społecznościowych, jak również procesów w nich zachodzących. Jednym z takich przykładów, który może być wykorzystany w procesie dydaktycznym lub naukowym, jest implementacja modelu Isinga (Cai, 2011), który doskonale odwzorowuje oddziaływanie poszczególnych opinii w strukturze sieci społecznościowej. Innym przykładem zastosowania może być implementacja algorytmu Girvana-Newmana (Girvan, 2002), która służy do wykrywania społeczności w jednolitej początkowo strukturze sieciowej poprzez sukcesywne usuwanie tzw. krawędzi między społecznościami. Przykład jego działania został zaprezentowany na rysunku 6.



Rysunek 6. Topologia wykorzystana do badania algorytmu Girvana-Newmana w środowisku Scilab (a) oraz wydzielone społeczności po dwóch cyklach działania algorytmu (b)

Źródło: opracowanie własne.

Należy stwierdzić, iż nie jest możliwe zaprezentowanie w jednej publikacji całego potencjału środowiska Scilab w rozważanym zakresie. Trzeba jednak zaznaczyć kilka zasadniczych kwestii:

Otwartość – jest to środowisko, które może, a wręcz powinno być rozwijane oraz rozbudowywane przez wszelkie instytucje tym zainteresowane. Szczególnie duże znaczenie ma to dla środowiska akademickiego, gdzie wkład jego przedstawicieli jest i może być nieoceniony w obszarze udostępniania efektywnego narzędzia dla pracy naukowców, a także do wykorzystania go w procesie dydaktycznym.

Darmowość – brak opłat związanych z jego wykorzystywaniem podczas badań oraz w trakcie nauczania jest jedną z jego dużych zalet, gdyż podobne rozwiązania komercyjne charakteryzują się wysokimi kosztami zakupu i utrzymania.

Elastyczność – modularność interfejsu wprowadza nielimitowaną możliwość konfiguracji wyglądu Scilab według potrzeb użytkownika. Poza podstawowymi komponentami modularność dotyczy również wszystkich narzędzi, np. SciNotes, Xcos czy pól rysowania wykresów. Dzięki temu rozwiązaniu praca może być zoptymalizowana i bardziej ergonomiczna.

Uniwersalność – prezentowane rozwiązanie posiada szeroki zakres zastosowań w różnych dziedzinach i specjalnościach, tzn. od prostych obliczeń matematycznych do optymalizacji numerycznej złożonych problemów i zagadnień inżynierskich, a nawet uczenia sztucznej inteligencji.

Podsumowanie

W artykule zaprezentowano możliwości wykorzystania środowiska Scilab w obszarze projektowania oraz modelowania sieci komputerowych. Obszar ten jest o tyle specyficzny, iż istnieje wiele przeznaczonych dla niego wyspecjalizowanych narzędzi komercyjnych. Jednak większość z tych środowisk związana jest ze stosunkowo wysokimi kosztami ich zakupu i utrzymania. Pamiętać należy też o tym, że są to rozwiązania zamknięte, uniemożliwiające ich rozwój i rozbudowę o dodatkowe moduły bez zgody producenta, natomiast proces dydaktyczny i projekty naukowe powinny cechować się jak największą otwartością na inne środowiska. Nie należy jednak zapomnieć, iż rozwijane w ten sposób środowisko nie jest wolne od drobnych błędów i występujących konfliktów pomiędzy modułami w przypadku najnowszych wersji oprogramowania. Sytuacje takie mają również miejsce w systemach tworzonych przez wiodących producentów oprogramowania.

Biorąc pod uwagę wbudowane możliwości funkcjonalne oprogramowania Scilab, a także dostępne moduły oferujące dodatkowe, wyspecjalizowane procedury, algorytmy i funkcje, można odpowiedzieć na zadane na początku artykułu pytanie, że środowisko to posiada duży potencjał i powinno znaleźć szerokie zastosowanie w zakresie modelowania i projektowania sieci komputerowych zarówno na poziomie dydaktycznym, jak i naukowym.

Literatura

- Albert, R., Barabási, A.L. (2002). Statistical Mechanics of Complex Networks. *Review of Modern Physics*, 74, 47–91.
- Albert, R., Jeong, H., Barabási, A.L. (1999). Diameter of the World-wide Web. *Nature*, 401, 130–131.
- Cai, W. (2011). *Handout 12. Ising Mode*. ME 334 Introduction to Statistical Mechanics, Stanford University.
- Girvan, M., Newman, M.E.J. (2002). Community Structure in Social and Biological Networks. *PNAS*, 99, 7821–7826.
- https://mars.uta.edu/mae3183/simulation/introscilab_baudin.pdf (20.01.2019).
- <https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/white-paper-c11-741490.html> (20.01.2019).
- Nicolis, G., Nicolis, C. (2012). *Foundations of Complex Systems: Emergence, Information and Prediction*. World Scientific Publishing Co.
- Sandep, N. (2017). *Introduction to Scilab*. Apress Berkeley.