

## MODEL SYMULACYJNY PROCEDUR DIAGNOSTYCZNYCH W ROZPROSZONYCH SYSTEMACH SIECIOWYCH

Zbigniew ZIELIŃSKI

Wojskowa Akademia Techniczna, Wydział Cybernetyki,  
Instytut Teleinformatyki i Automatyki, ul. Kaliskiego 2, 00-908 Warszawa; fax: 022-683-71-44;  
e-mail: [z.zielinski@ita.wat.edu.pl](mailto:z.zielinski@ita.wat.edu.pl)

### Streszczenie

W artykule przedstawiono założenia i model symulacyjny procesu diagnostyki w sieciowych systemach rozproszonych, pozwalający na uzyskanie oceny efektywności procesu diagnozowania dla zadanej struktury diagnostycznej i parametrów niezawodnościowych systemu. Przedstawiono również w zarysie, sposób implementacji modelu oparty na podejściu obiektowym.

Słowa kluczowe: samodiagnozowalne sieciowe systemy rozproszone, tolerowanie błędów, symulacja.

### THE SIMULATION MODEL OF DISTRIBUTED NETWORK SYSTEM DIAGNOSTIC PROCEDURES

#### Summary

In this paper the simulation model of distributed network system diagnosing procedures is presented. The main goal of the model is effectiveness evaluation of the diagnostic procedures depending on communication and diagnostic structure parameters and reliability node characteristics. More over, the outline of the model implementation is depicted.

Keywords: self-diagnosable distributed systems, fault tolerance, simulation.

### WPROWADZENIE

Postęp w dziedzinie technologii sieci komputerowych, a zwłaszcza gwałtowny rozwój sieci bezprzewodowych, doprowadził do wzrostu znaczenia systemów o charakterze rozproszonym. Coraz częściej obserwujemy wykorzystanie systemów rozproszonych w obszarach o tzw. zastosowaniach krytycznych (systemy militarne, lotnicze, medyczne itp.). Powoduje to określony wzrost wymagań, dotyczących wiarygodności systemów rozproszonych.

Zwiększenie wiarygodności rozproszonych systemów komputerowych może być uzyskane na drodze stosowania efektywnych metod diagnozowania i odtwarzania stanu po awarii, a w szczególności, metod wykorzystujących rezultaty wzajemnego testowania się komputerów. Systemy zdolne do identyfikacji niezdatnych jednostek należących do systemu zalicza się do samodiagnozowalnych. W [12] dokonano klasyfikacji struktur samodiagnozowalnych zależnie od metody i sposobu interpretacji wyników poszczególnych testowań, jak i modelu wnioskowania o stanie niezawodnościowym. Podzielono je na struktury dialogu diagnostycznego (DD) i struktury opiniowania diagnostycznego (OD). W przypadku struktur DD, wnioskowanie o stanie niezawodnościowym systemu, odbywa się na podstawie wyników części wszystkich

możliwych testowań, a w strukturach OD przeciwnie.

Biorąc pod uwagę sposób testowania oraz przywracania zdatności systemu, wyróżnia się dwie strategie diagnozowania [7, 10]: strategia jednokrokowa, strategia wielokrokowa (sekwencyjna). Strategia jednokrokowa polega na wykonaniu wszystkich dopuszczalnych testów w systemie i wyznaczeniu wszystkich uszkodzonych jednostek na podstawie otrzymanego zbioru wyników testowań (opinii globalnej). W przypadku strategii wielokrokowej proces diagnozy i odtwarzania systemu przeplatają się nawzajem. Przyjmuje się, że na podstawie opinii globalnej jesteśmy w stanie określić tylko pewien niepusty podzbiór uszkodzonych jednostek (na poziomie architektury sieci). Następnie zidentyfikowane uszkodzone jednostki wymienia się na zdadne i ponawia się testowanie.

Jakość diagnostyki ma decydujące znaczenie dla przywrócenia zdatności systemu poprzez wymianę uszkodzonych jednostek, bądź też izolację niezdatnych jednostek i przeprowadzenie rekonfiguracji zadań. Należy zaznaczyć, że zwiększenie odporności na błędy w systemie rozproszonym jest znacznie trudniejsze niż w systemie scentralizowanym (wieloprocesorowym).

Zwykle w systemach o zwiększonej odporności na uszkodzenia bierze się pod uwagę dwie strategie

diagnostyczne, pierwsza dotyczy diagnozy dokładnej (jednoznacznej), druga związana jest z diagnozą nadmiarową. W przypadku systemów jednoznacznie diagnozowalnych chodzi o precyzyjną identyfikację wszystkich niezdatnych jednostek (komputerów) w systemie na podstawie uzyskanych wyników testów wzajemnych tych jednostek przy założeniu, że liczba niezdatnych jednostek systemu nie przekroczy ustalonego parametru  $m$ .

W literaturze z obszaru diagnostyki systemowej istnieje bardzo duża opracowanych modeli i metod diagnostycznych. Odnoszą się one zarówno do systemów wieloprocesorowych jak i w pewnej części do systemów rozproszonych. Należy zauważyć, że w większości przypadków modele te mają charakter czysto teoretyczny, tj. przyjęte założenia istotnie ograniczają możliwości ich praktycznego wykorzystania.

Najczęściej przyjmowane założenia nie ujmują problematyki dotyczącej skuteczności kontrolnej testów. Zazwyczaj w prowadzonych rozważaniach nie wnika się w fizyczną naturę wykonywanych testów i traktowane są one jako testy kompletne.

Kolejnym ograniczeniem większości metod diagnostycznych jest brak uwzględniania różnych rodzajów niezdatności (trwałe, powtarzające się i chwilowe) oraz losowej natury niezdatności. W przypadku systemów rozproszonych niezdatności mogą dotyczyć oprogramowania, węzłów przetwarzających jak i linii transmisji danych. W literaturze przedmiotu najczęściej przyjmuje się założenie, że linie transmisji danych są niezawodne. Nieliczne publikacje, na przykład, [12] dotyczą analizy systemów z zawodnymi liniami transmisji danych.

W rzeczywistości uwzględnienie wszystkich wymienionych czynników znacznie komplikuje model matematyczny diagnostyki systemu rozproszonego i stwarza określone trudności w jego analitycznym ujęciu. Znaczny stopień komplikacji wynikający z uwzględniania wielu czynników diagnostyki systemu rozproszonego, a także takie cechy współczesnych systemów sieciowych jak znaczna niejednorodność (łączone są systemy o różnych technologiach i możliwościach funkcjonalnych oraz operacyjnych), zmienność struktury itp., skłaniają do wykorzystania badań symulacyjnych. Przegląd istniejących rozwiązań w zakresie symulacji sieciowych systemów rozproszonych przedstawiono w [4]. Rozwiązania te w większości wykorzystują symulację funkcjonalną i nie obejmują modelu procesów diagnostycznych.

Podstawowym celem modelu symulacyjnego jest uzyskanie oceny efektywności procesu diagnozowania dla zadanej struktury diagnostycznej i parametrów niezawodnościowych systemu rozproszonego. Pozwoli to projektantowi takiego systemu na dobór odpowiednich parametrów niezawodnościowych węzłów i linii

komunikacyjnych oraz strategii i metod diagnostycznych dla uzyskania wymaganego poziomu wiarygodności systemu rozproszonego.

W artykule przedstawiono założenia i model symulacyjny procesu diagnostyki w sieciowych systemach rozproszonych. Ponadto w pracy zaprezentowano podstawowe elementy wykorzystywane w implementacji modelu.

## 2. ZAŁOŻENIA DO MODELU SYMULACYJNEGO

Niech  $E = \{e_i : i = \overline{1, k}\}$  reprezentuje zbiór węzłów systemu rozproszonego. Strukturę komunikacyjną systemu przedstawia się za pomocą grafu  $C (C = \langle E, U \rangle)$ , w którym węzły odpowiadają komputerom, a zbiór możliwych łańcuchów prostych, łączących węzeł  $e_i$  z węzłem  $e_j (i \neq j)$ , odpowiada zbiorowi możliwych sposobów przekazania informacji z komputera  $e_i$  do komputera  $e_j$ . Sieć komputerową, przy określonym stanie niezawodnościowym jej linii transmisji danych, uważamy za zdatną, jeżeli graf  $C (C = \langle E, U \rangle)$  opisujący jej strukturę logiczną, przy tym stanie niezawodnościowym linii transmisji danych, jest grafem spójnym [1].

Każde łącze komunikacyjne może być charakteryzowane przez prawdopodobieństwo błędnego przesłania komunikatu ( $\delta$ ). Wewnętrzne struktury danych reprezentujące strukturę komunikacyjną są następujące:

- liczba węzłów  $k = |E|$ ,
- $\frac{1}{2}$  macierzy  $[c_{ij}]$ ,  $(k \times k)$ ,
- $\frac{1}{2}$  macierzy  $[\delta_{ij}]$ ,  $(k \times k)$ .

Strukturę wzajemnego testowania się elementów systemu opisuje odpowiedni graf (*graf testowania*). Grafem testowania  $D (D = \langle E, T \rangle)$ , jest (w tym przypadku) unigraf bez pętli, którego gałęzie są łukami, przy czym łuk  $(e', e'') \in T$  oznacza, że węzeł  $e'$  może testować węzeł  $e''$ .

Zależnie od rodzaju zastosowanego modelu diagnostycznego, wyniki wzajemnego testowania się komputerów w systemie o scentralizowanym sposobie diagnozowania można przedstawić za pomocą podanych poniżej zależności. Największe znaczenie praktyczne znalazły modele PMC [7], [10], [12], [13] oraz BGM [2], [7], [10], [12].

W przypadku przyjęcia modelu PMC interpretacja wyników testowania przedstawiana jest za pomocą następujących zależności [10]:

$$[n(e_s) = n_0(e_s)] \Rightarrow \left[ d_{st} = \begin{cases} 0 & \text{dla } n(e_t) = n_0(e_t) \\ 1 & \text{dla } n(e_t) \neq n_0(e_t) \end{cases} \right] \quad (1)$$

$$\text{oraz: } [n(e_s) \neq n_0(e_s)] \Rightarrow [d_{st} = x] \quad (x \in \{0, 1\})$$

gdzie  $n(e_s)$  – stan niezawodnościowy komputera, który w grafie reprezentowany jest przez węzeł  $e_s$ ;

$n_0(e_s)$  – stan zdadności komputera  $e_s$ ;

$d_{st}$  – opinia komputera testującego  $e_s$  o komputerze testowanym  $e_t$ , przy czym  $d_{st} = 0$  oznacza, że komputer  $e_s$  opiniuje jako zdadny komputer  $e_t$ , natomiast  $d_{st} = 1$  oznacza, że komputer  $e_s$  opiniuje jako niezadny komputer  $e_t$ .

Jeżeli przyjęty zostanie model BGM, to wówczas wynik testowania się komputerów przedstawiany jest za pomocą następującej zależności [10]:

$$[n(e_s) \neq n_0(e_s)] \Rightarrow \left[ d_{st} = \begin{cases} x & \text{dla } n(e_t) = n_0(e_t) \\ 1 & \text{dla } n(e_t) \neq n_0(e_t) \end{cases} \right]. \quad (2)$$

W odróżnieniu od scentralizowanych metod diagnostyki, zarówno interpretacja wyników testów jak i procedura określania decyzji diagnostycznej w systemach o rozproszonym sposobie diagnostyki mogą być bardziej złożone. W literaturze przedmiotu scharakteryzowano rozproszone sposoby (metody) diagnozowania systemów sieciowych, do których zaliczyć można metodę dialogu diagnostycznego przedstawioną w [10] i [12], metody adaptacyjne scharakteryzowane w [15] oraz tzw. zintegrowane metody diagnostyki w systemie rozproszonym [8], [9].

W opracowywanym modelu symulacyjnym uwzględniono zarówno podejście scentralizowane jak i rozproszone sposoby diagnozowania systemu.

Niezależnie od zastosowanego podejścia (scentralizowane czy rozproszone) założono, że sposób wzajemnego testowania się komputerów oraz parametry testów definiuje graf opisany  $D^* = \langle D; \{\eta(t_{ij}), \tau(t_{ij})\}; t_{ij} \in T \rangle$  gdzie  $\eta(t_{ij})$  – jest skutecznością kontrolną testu  $t_{ij}$ ,  $\tau(t_{ij})$  jest czasem wykonania testu  $t_{ij} = \langle e_i, e_j \rangle \in T$ .

Skutecznością kontrolną  $\eta(t_{ij})$  testu  $t_{ij}$  ( $t_{ij} \in T$ ) nazywamy wartość wyrażenia [10]:

$$\eta(t_{ij}) = \frac{1}{1-p_0} P(r(t_{ij}) \neq r(t_{ij}, n_0)) \quad (0 \leq \eta(t_{ij}) \leq 1) \quad \text{gdzie}$$

$p_0 = P(n(e_i) = n_0)$  ( $0 < p_0 < 1$ ),  $r(t_{ij})$  oznacza reakcję (odpowiedź) komputera  $e_j$  na test zlecony przez komputer  $e_i$ , a  $r(t_{ij}, n_0)$  jest odpowiedzią  $e_j$  w stanie zdadności.

Kolejnymi elementami wprowadzonymi do modelu symulacyjnego są parametry niezawodnościowe węzłów, rodzaje niezadności oraz zachowanie węzłów uszkodzonych. Stan niezawodnościowy każdego węzła  $e_i$  będzie określony przez zmienną losową  $X(e_i)$  o zadanej dystrybucji ( $P[X(e_i) < x]$ ). W modelu założono

obsługę następujących rozkładów zmiennej losowej  $X(e_i)$ :

- rozkład wykładniczy,
- rozkład Weibull'a..

W modelu uwzględniono dwa rodzaje niezadności węzła: trwałe i przemijające (okresowe). W przypadku uwzględnienia niezadności przemijających charakteryzowane są za pomocą odpowiednich rozkładów prawdopodobieństwa.

Węzły niezadne (dla niezadności trwałych) mogą być wyłączane z systemu lub poddawane procesowi naprawy. Proces naprawy węzła charakteryzowany jest przez średni czas naprawy węzła (MTTR - *mean time to repair*). Założono w modelu, że domyślna wartość MTTR węzła wynosi "nieskończoność". Oznacza to brak naprawy i wyłączenie węzła z systemu po identyfikacji w tym węzle stanu niezadności. Istnieje możliwość charakteryzowania wartości MTTR za pomocą rozkładu normalnego (określana jest wartość średnia i odchylenie standardowe). Istnieje również możliwość uwzględnienia takiego scenariusza uzdrawiania systemu, gdzie po wykryciu niezadnego węzła zastępowany jest on innym (zadnym) węzłem i proces diagnozy powtarzany jest dopóty, dopóki wszystkie węzły niezadne nie zostaną usunięte z systemu.

Jeżeli zachowanie uszkodzonego węzła zostało określone jako uciszające (*fail-silent*), to przyjmuje się, że węzeł nie odpowiada na żadne komunikaty (nie bierze udziału ani w przetwarzaniu ani w procesie diagnozowania). Sposób konstrukcji węzłów typu *fail-silent* przedstawiono w pracy [3]. Sprowadza się on do zastosowania w pojedynczym węzle dwóch procesorów kontrolujących wzajemnie swoje działania z wykorzystaniem specjalnego protokołu. Zachowanie bizantyjskie oznacza przypadkowość zachowania węzła.

Każdy węzeł może być charakteryzowany przez następujące atrybuty:

- typ węzła (prosty, wirtualny),
- zachowanie węzła po wystąpieniu uszkodzenia (uciszające - *fail-silent*, bizantyjskie),
- model uszkodzeń..

Koncepcja węzłów wirtualnych została przedstawiona w [9]. Polega ona na logicznym wyodrębnieniu grup kilku jednostek fizycznych (węzłów prostych) o podobnych parametrach. Zadanie rozproszone jest realizowane jednocześnie we wszystkich jednostkach wchodzących w skład węzła. Poprawny wynik jest ustalany na drodze głosowania. Wynik ten jest następnie przekazywany „na zewnątrz” jako poprawny wynik całego węzła wirtualnego.

### 3. METODA SYMULACJI

Modelowany system rozproszony działa w czasie ciągłym, natomiast zmiany stanu systemu,

istotne z punktu widzenia celu modelowania (procedur diagnostycznych) zachodzą w dyskretnym (przeliczalnym) zbiorze chwil czasowych. Zmiany stanu systemu, zachodzące w dyskretnych chwilach czasu, określa się jako zdarzenia. Stany niezawodnościowe węzłów systemu rozproszonego i linii transmisji danych oraz zachowanie węzłów uszkodzonych mają charakter losowy. Stąd, do badania sposobu diagnozowania systemu rozproszonego wykorzystana została metoda symulacji stochastycznej z dyskretnym zbiorem stanów.

Każde zdarzenie  $\varepsilon_i$  charakteryzowane jest przez uporządkowaną czwórkę:

$$\varepsilon_i = \langle z_i, \tau_i, \rho_i, P_i \rangle$$

gdzie  $z_i$  jest identyfikatorem zdarzenia,  $\tau_i$  jest chwilą czasu, w której zachodzi zdarzenie  $\xi_i$ ,  $\rho_i$  oznacza typ zdarzenia (wyróżnionymi typami zdarzeń są, na przykład, zmiana stanu niezawodnościowego węzła - *fault*, przesłanie komunikatu, błąd przesłania komunikatu itp.), a  $P_i$  jest wektorem parametrów zdarzenia. Lista zdarzeń  $\kappa = (\xi_i, i = 1, l)$  uporządkowana niemalejąco według czasu wystąpienia zdarzenia nosi nazwę kalendarza zdarzeń. W modelu wykorzystuje się dwie klasy metod obsługi zdarzeń: metoda przeglądania możliwych zdarzeń oraz metoda procesów interakcyjnych, w której każdy z wyróżnionych obiektów modelu wykonuje wspólnie swoje operacje.

Ogólny algorytm symulacji, realizujący metodę przeglądania możliwych zdarzeń przedstawia rys. 1.

<b>Krok 1. (Inicjalizacja modelu)</b>
<p>a. Wprowadź strukturę komunikacyjną <math>C</math> oraz diagnostyczną <math>D^*</math> i ich parametry;</p> <p>b. Wprowadź atrybuty i parametry dotyczące węzłów;</p> <p>c. Wybierz procedury diagnostyczne oraz określ warunek zakończenia symulacji (<math>\alpha_{SIM}</math>);</p> <p>d. Inicjuj współbieżne wątki węzłów.</p>
<b>Krok 2. (Wykonanie symulacji)</b>
<p>POWTARZAJ dopóki <math>\alpha_{SIM} \neq \text{PRAWDA}</math></p> <p>{a. Znajdź zdarzenie <math>\xi_i \in \kappa</math> typu <i>fault</i> z minimalnym czasem wystąpienia <math>\tau_i</math>;</p> <p>b. Uruchom procedury diagnostyczne w przedziale czasowym (<math>\tau_i - T_{GLOB}</math>);</p> <p>c. Zmień wartość czasu globalnego na <math>\tau_i</math> (<math>T_{GLOB} = \tau_i</math>);</p> <p>d. Rejestruj wyniki diagnozy.}</p>
<b>Krok 3. (Estymacja wybranych wskaźników i zobrazowanie wyników)</b>

Rys. 1 Ogólny algorytm symulacji

W każdym obiekcie modelu typu węzeł wykonywany jest również współbieżnie algorytm symulacji stanu niezawodnościowego i rodzaju powstałych niezdatności.

Zależnie od symulowanej procedury diagnostycznej estymowane są następujące parametry:

- $\gamma_{diag}$  - stosunek liczby poprawnych decyzji diagnostycznych do liczby wszystkich decyzji diagnostycznych,
- $\Delta(t)$  - prawdopodobieństwo detekcji jako funkcja czasu,
- $\gamma_{nap}$  - średnia liczba kroków diagnozy i naprawy niezbędna do przywrócenia stanu pełnej zdadności systemu (dla strategii wielokrokowej),
- $\Lambda$  - średni czas opóźnienia diagnozy (dla procedur rozproszonego diagnozowania systemu). Opóźnienie diagnozy oznacza czas upływający od chwili pojawienia się niezdatności do chwili przekazania informacji diagnostycznej o tej niezdatności do wszystkich węzłów systemu.

#### 4. IMPLEMENTACJA MODELU

Do implementacji modelu wykorzystano paradygmat obiektowy realizacji oprogramowania oraz interfejs graficzny. Zrealizowane w języku C++ oprogramowanie systemu symulacji o nazwie **SIMDIAG** zostało podzielone na trzy współpracujące ze sobą komponenty o następujących nazwach:

- **Budowniczy modelu,**
- **Symulator,**
- **Moduł zobrazowania wyników symulacji.**

Moduł o nazwie **Budowniczy modelu** umożliwia:

- definiowanie struktury komunikacyjnej,
- definiowanie struktury diagnostycznej,
- określanie parametrów węzłów przetwarzających,
- specyfikowanie procedur diagnostyki systemu rozproszonego.

Definiowanie struktury komunikacyjnej odbywa się z wykorzystaniem interfejsu graficznego. Interfejs ten dostarcza elementarnych operacji typu: dodaj węzeł, usuń węzeł, dodaj połączenie między węzłami, usuń połączenie, wybierz element graficzny (węzeł, połączenie), wyświetl atrybuty wybranego elementu.

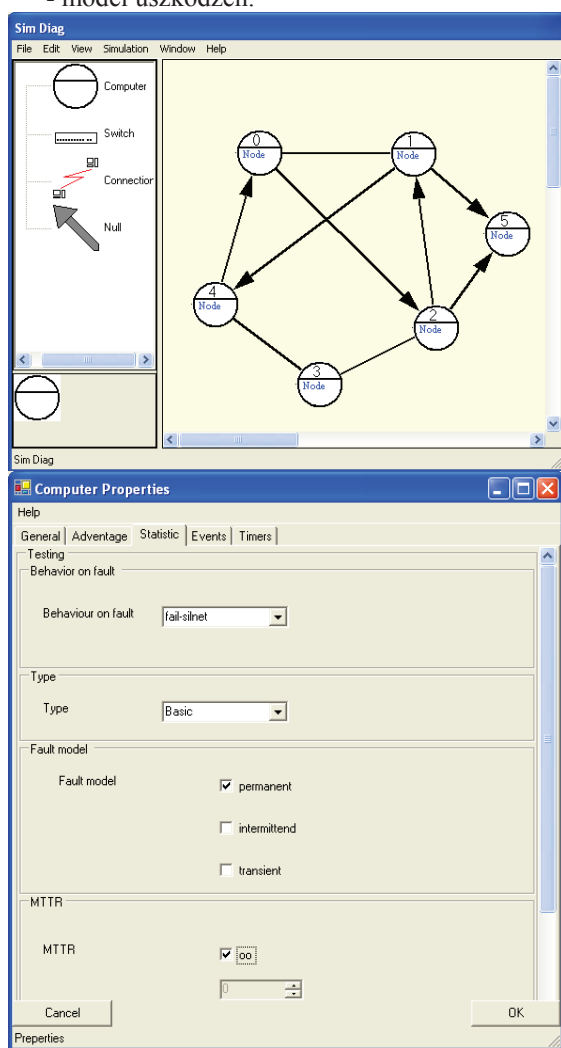
Definiowanie struktury diagnostycznej następuje po uprzednim określeniu struktury komunikacyjnej  $C$ . Program umożliwia wstępne zobrazowanie struktury komunikacyjnej (zaznaczając linie transmisji danych kolorem szarym) oraz umożliwia wprowadzanie testów poprzez interfejs graficzny. Testy są reprezentowane w postaci łuków bądź krawędzi grafu. Elementarnymi operacjami w tym zakresie są operacje typu: dodaj test, usuń test, wybierz test, podaj (zmień) atrybuty wybranego testu. Każdy test

$t_{ij}$  jest charakteryzowany przez: skuteczność kontrolną testu  $\eta_{ij}, (0 \leq \eta_{ij} \leq 1)$ , liczbę komunikatów wysyłanych (związanych z testem  $t_{ij}$  -  $l(t_{ij})$ ) oraz średni czas wykonania testu  $\tau_{ij}$ .

Wprowadzone parametry modelu zapisywane są do pliku typu XML, co umożliwia elastyczną interpretację modelu w komponencie *Symulator* bądź jego późniejszą modyfikację. Przykład definiowania struktury diagnostycznej oraz postać interfejsu graficznego przedstawiono na rys. 2.

Każdy węzeł może być charakteryzowany przez:

- typ węzła (prosty, wirtualny),
- zachowanie węzła po wystąpieniu uszkodzenia (*fail-silent*, bizantyjskie),
- model uszkodzeń.



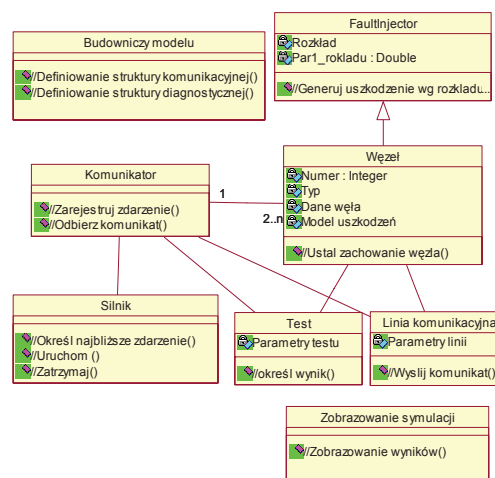
Rys. 2. Przykład definiowania struktury diagnostycznej

Węzły zidentyfikowane w wyniku zastosowania określonych procedur diagnostycznych jako niezdatne mogą być wyłączane z systemu lub poddawane procesowi naprawy. Wykorzystuje się tu parametr węzła - średni czas naprawy (MTTR - *mean time to repair*).

Do podstawowych klas obiektów komponentu *Symulator* możemy zaliczyć (rys. 3):

- *Węzeł*,
- *FaultInjector*,
- *Test*,
- *Linia komunikacyjna*,
- *Komunikator*,
- *Silnik*.

Należy podkreślić (zgodnie z rys. 3), że klasa *FaultInjector* opisuje jeden z podstawowych obiektów symulatora *SIMDIAG*, obejmujący mechanizm wprowadzania niezdatności w symulowanym systemie. Mechanizm ten służy zarówno do wyznaczania chwil czasowych zdarzeń powodujących zmianę stanu niezawodnościowego węzła jak i do określania rodzaju symulowanej niezdatności.



Rys. 3. Struktura oprogramowania – diagram wybranych klas

Specyfikacja procesu diagnostycznego wymaga określenia: strategii diagnostycznej, sposobu wyznaczania i interpretacji wyników testów oraz procedur (algorytmów) diagnostycznych. W modelu mogą być symulowane następujące procedury [10], [15]:

- scentralizowana procedura opiniowania diagnostycznego,
- rozproszona procedura zorientowana na zdarzenia (*EVENT-SELF*),
- adaptacyjna procedura DSD (*Distributed System Diagnosing*),
- procedura dialogu diagnostycznego.

W przypadku symulacji scentralizowanego sposobu diagnozowania systemu zakłada się, że istnieje węzeł o zwiększonych parametrach niezawodnościowych, posiadający możliwość niezawodnej komunikacji z pozostałymi węzłami. W tym przypadku węzeł nadrzędny koordynuje wykonywanie testów. Implementacja węzła nadrzędnego jest zawarta w klasie *Komunikator* (rys. 3). Wszystkie węzły systemu rozproszonego wykonują założone testy (z określonym przez

parametr interwałem), a wyniki testów przesyłane są do węzła nadrzędnego.

Oprogramowanie *SIMDIAG* umożliwia zobrazowanie (w postaci graficznej) zarówno wartości chwilowych wybranych parametrów diagnostycznych jak i ich estymatorów (wartość średnia, wariancja).

## 5. PODSUMOWANIE

Przedstawiony w artykule model symulacyjny daje możliwości badania wpływu szeregu parametrów, określających właściwości struktury komunikacyjnej i diagnostycznej, charakterystyk niezawodnościowych węzłów przetwarzających, rodzaju niezdatności oraz sposobu diagnozowania na efektywność procedur diagnozowania rozproszonego systemu sieciowego.

W chwili obecnej prowadzone są testy końcowe oprogramowania *SIMDIAG* oraz dokonywana jest ocena adekwatności uzyskiwanych wyników symulacji.

## LITERATURA

- [1] Chudzikiewicz J., Zieliński Z.: *Wyznaczanie m-diagnostycznych struktur typu PMC w systemach o zwiększonej odporności na uszkodzenia*, Materiały X Konferencji Systemów Czasu Rzeczywistego, Ustroń, 2003.
- [2] Barsi F., Grandoni F., Maestrini P.: *A Theory of Diagnosability of Digital Systems*, IEEE Trans. on Comput. 6, 1976, pp. 585-593.
- [3] Brasileiro F. V., Ezhilchelvan P. D., Shrivastava S. K.: *Implementing Fail Silent Nodes for Distributed Systems*, IEEE Trans. on Computers vol. 45, no 11, 1996, pp. 1226-1238.
- [4] Goswami K. K., Iyer R. K., Young L.: *DEPEND: A Simulation-Based Environment for System Level Dependability Analysis*, IEEE Trans. on Computers, Vol. 46, No 1, 1997, pp. 60-74.
- [5] Hakimi S. L., Amin A. T.: *Characterization of Connection Assignment of Diagnosable Systems*, IEEE Trans. on Computers, 1, 1974, pp. 86-88.
- [6] Hosseini S. H., Kuhl J. G., Reddy S. M.: *On Self-fault diagnosis of the distributed systems*. IEEE Trans, Computers, vol. 37, pp. 248-251, Feb. 1988.
- [7] Krawczyk H.: *Analiza i synteza samodiagnostycznych systemów komputerowych*, Zeszyty Naukowe Politechniki Gdańskiej, Elektronika nr 64, Gdańsk, 1987.
- [8] Krawczyk H., Umiński P. W.: *Diagnozowanie i odtwarzanie poprawnego stanu w rozproszonych systemach sieciowych*, Zeszyty naukowe Politechniki Śląskiej, Informatyka, zeszyt 28, Gliwice, 1995.
- [9] Krawczyk H., Umiński P. W.: *Węzły wirtualne jako metoda zwiększania odporności na błędy aplikacji działających w systemach rozproszonych*, Materiały konferencji Real-Time Systems' 95, OWPW, Wrocław, 1995.
- [10] Kulesza R.: *Podstawy diagnostyki sieci logicznych i komputerowych*, Instytut Automatyki i Robotyki, Wydział Cybernetyki WAT, Wydanie II, Warszawa, 2000.
- [11] Kulesza R.: *Problemy przeliczania optymalnych struktur opiniowania diagnostycznego*, Biuletyn Instytutu Automatyki i Robotyki nr 20/2004, Warszawa.
- [12] Kulesza R.: *Struktury samodiagnostyczne w systemach cyfrowych*, Materiały Krajowej Konferencji DIAG' 2003.
- [13] Preparata F. P., Metze G., Chien R.T.: *On the Connection Assignment Problem of Diagnosable Systems*, IEEE Trans. Comput. 6, 1967, pp. 848-854.
- [14] Somani A. K., Peleg O.: „On diagnosability of Large fault Sets and Its Applications to Regular-Interconnected Computer Systems”, IEEE Trans on Computers, Vol 45, No. 8, 1996, pp. 892-903.
- [15] Stahl M., Buskens R., Bianchini R. Jr.: *On-Line Diagnosis In General Topology Networks*, Proc. of the Fault-Tolerant and Distributed Systems, Amherst, MA, July 1992.
- [16] Zieliński Z.: *Komputerowo wspomagane wyznaczanie najtańszych 2-diagnostycznych struktur typu PMC*, Biuletyn Instytutu Automatyki i Robotyki nr 18/2003, Warszawa.



Dr inż. **Zbigniew ZIELIŃSKI** ukończył z wyróżnieniem studia na Wydziale Cybernetyki Wojskowej Akademii Technicznej w 1978 roku. W roku 1988 obronił pracę doktorską uzyskując tytuł doktora nauk technicznych

specjalności systemy informatyczne. Przez cały okres swojej pracy zawodowej związany z Wojskową Akademią Techniczną. Brał udział w wielu pracach naukowo-badawczych związanych z projektowaniem i diagnozowaniem systemów wbudowanych. Aktualnie jego zainteresowania naukowe dotyczą metod diagnozowania systemów i sieci komputerowych oraz projektowania systemów tolerujących błędy.