Twierdzenie o schematach

Próbą wyjaśnienia dużej skuteczności algorytmów genetycznych w roli uniwersalnej metody odnajdywania optymalnego rozwiązania bez znajomości charakterystyki zadania jest koncepcja schematu. Schemat jest to wzorzec opisujący podzbiór ciągów podobnych ze względu na ustalone pozycje. Użycie takiej definicji schematu wymaga dodania do alfabetu genów specjalnego symbolu * (nieistotne) reprezentującego wartość nieokreśloną. Schemat reprezentuje wszystkie łańcuchy będące podzbiorem przestrzeni poszukiwań, które zgadzają się z nim na wszystkich pozycjach innych niż *.

Pojęcie schematu jest obowiązujące dla alfabetu genów o dowolnej długości. Przy założeniu, że ciągi kodowe reprezentujące genotypy składają się z symboli alfabetu dwójkowego, na każdej pozycji schematu może pojawić się każdy symbol należący do zbioru V = { 0, 1, * }. Na przykład schemat H1 = (*111100100) złożony z symboli alfabetu V może być reprezentowany przez dwa następujące ciągi:

{ (0111100100),(1111100100) }

natomiast reprezentantami schematu H2 = (*1*1100100) są wyłącznie cztery następujące ciągi:

{ (0101100100),(0111100100),(1101100100),(1111100100)}.

Każdy schemat reprezentowany jest przez 2^r ciągów binarnych, gdzie r jest liczbą symboli nieistotne * w szablonie schematu. Z drugiej strony każdy łańcuch binarny o długości m jest reprezentantem 2^m schematów. Na przykład łańcuch binarny (101001) jest reprezentantem 26 schematów:

(1 0 1 0 0 1)

(* 0 1 0 0 1)

(1 * 1 0 0 1)

(1 0 1 0 0 *) (* * 1 0 0 1) (* 0 * 0 0 1) (1 0 1 0 * *) (* * * 0 0 1)

(* * * * * *

Dla łańcuchów o długości m istnieje 3^m możliwych schematów. W populacji o liczebności n może być reprezentowanych od 2^m do n x 2^m różnych schematów. Już te proste obliczenia dają pewne wyobrażenie o ogromie informacji przetwarzanej przez algorytmy genetyczne; aby jednak zrozumieć schematy – cegiełki, z których powstają przyszłe rozwiązania trzeba wyodrębnić różne typy schematów. Do sformułowania twierdzenia o schematach potrzebne będą dwa atrybuty charakteryzujące schemat: rząd i rozpiętość nazywana również długością definiującą.

Rzędem schematu H, oznaczonym przez o(H) jest nazywana liczba ustalonych pozycji we wzorcu. Dla ciągów binarnych jest to liczba zer i jedynek. Np. o(011*1**) = 4 oraz o(0******) = 1.

Rozpiętość schematu H, którą oznaczamy przez 5(H), to inaczej odległość między dwiema skrajnymi pozycjami ustalonymi. Tak więc, schemat (011*1**) ma rozpiętość równą 4, gdyż jego ostatnia ustalona pozycja ma numer 5, a pierwsza – numer 1, zatem odległość między nimi wynosi 5 – 1 = 4. Schemat który ma tylko jedną ustaloną pozycję, taki jak (0*******), ma rozpiętość 5 = 0.

Pierwszym i najłatwiejszym elementem teorii schematów jest ustalenie wpływu selekcji, czy inaczej reprodukcji na oczekiwaną liczbę reprezentantów określonego schematu. Wyrażenie m = m(H, t) oznacza liczbę reprezentantów schematu H w chwili t. Podczas selekcji genotypy trafiają do puli rodzicielskiej z prawdopodobieństwem proporcjonalnym do wskaźnika przystosowania, a dokładnie z prawdopodobieństwem równym:

$$p_i = f_i / \sum f_i \quad (1.4.)$$

Po skompletowaniu puli rodzicielskiej nowego pokolenia złożonej z n ciągów wybranych w chwili t, można oczekiwać obecności m = m(H, t +1) reprezentantów schematu H w chwili t +1, przy czym zachodzi zależność:

$$E[m(H,t+1)] = m(H,t) \cdot n \cdot f(H) / \sum f_i \quad (1.5.)$$

Liczba f(H) określa średnie przystosowanie ciągów będących reprezentantami schematu H w chwili t. Po uwzględnieniu ilorazu sumy wartości przystosowania wszystkich chromosomów populacji i ich liczby jako średniego przystosowania całej populacji t oraz pominięciu symbolu wartości oczekiwanej po lewej stronie równania 1.5. otrzymuje się słuszną dla dużych populacji aproksymację:

$$m(H,t+1) = m(H,t) \frac{f(H)}{\overline{f}}$$

Oznacza to, że liczebność reprezentacji danego schematu w następnym pokoleniu zmienia się proporcjonalnie do stosunku średniego przystosowania schematu i średniego przystosowania całej populacji; a zatem schematy o przystosowaniu wyższym niż średnie populacji będą miały większą liczbę reprezentantów w następnym pokoleniu, podczas gdy schematy o przystosowaniu niższym niż średnie z populacji otrzymają mniej reprezentantów niż miały dotychczas.

Przy założeniu, że pewien schemat H przewyższa średnią o jakąś jej stałą część równą c, równanie schematów można zapisać tak jak na wzorze 1.7., a startując od t=0 i zakładając, że c nie zmienia się w czasie, otrzymuje się zależność przedstawiona na wzorze

$$m(H,t+1) = m(H,t)(\overline{f} + c\overline{f})/\overline{f} = (1+c)\cdot(H,t)$$

$$m(H,t) = m(H,0)\cdot(1+c)^t \qquad (1.7.)$$

(1.8.)

Otrzymana ocena ilościowa efektu reprodukcji pozwala stwierdzić, że schematy lepsze od przeciętnej są wybierane w liczbie przypadków rosnącej wykładniczo w czasie, natomiast gorsze od przeciętnej wybierane są w liczbie wykładniczo malejącej w czasie.

Kolejnym krokiem jest zbadanie wpływu operatora krzyżowania na oczekiwaną liczbę reprezentantów określonego schematu. Operacja krzyżowania dokonuje uporządkowanej wymiany informacji pomiędzy dwoma chromosomami. Przy pomocy krzyżowania powstają nowe struktury w sposób minimalnie zaburzający wzorzec propagacji dyktowany przez reprodukcję. W końcowym efekcie frekwencje różnych schematów reprezentowanych w populacji rosną lub maleją wykładniczo w kolejnych pokoleniach.

Oszacowanie wrażliwości schematu na krzyżowanie rozpocząć można od rozważenia przykładowego ciągu kodowego A o długości l = 33 oraz dwóch reprezentatywnych schematów H1 i H2 , pasujących do tego ciągu:

A = (11101111101000100011000000100011 0)
H1 = (****111**************************)
H2 = (111**************************10)

Przy założeniu, że łańcuch został wybrany do krzyżowania i losowo wybrano punkt cięcia na pozycji 20, łatwo można zauważyć, iż schemat H1 przetrwał krzyżowanie, czyli że jeden z potomków ciągu A nadal jest reprezentantem schematu, natomiast drugi schemat nie przetrwał. Przyczyna leży w tym, że punkt cięcia w pierwszym schemacie nie trafił pomiędzy

ustalone pozycje, natomiast ustalone pozycje 111 na początku szablonu H2 oraz 10 na jego końcu znalazły się w różnych potomkach. Jest oczywiste, że rozpiętość schematu spełnia istotną rolę w prawdopodobieństwie jego przetrwania lub zniszczenia.

Dla każdego schematu można podać dolne oszacowanie prawdopodobieństwa przeżycia podczas krzyżowania. Kiedy punkt krzyżowania nie trafia pomiędzy ustalone pozycje schemat przeżywa krzyżowanie, więc prawdopodobieństwo przeżycia ps wynosi co najmniej 1 – 5(H)/(l-1) . Jeśli dla określonej pary ciągów sama operacja krzyżowania jest wykonywana z pewnym prawdopodobieństwem pc, to prawdopodobieństwo przeżycia schematu H spełnia nierówność:

$$p_s = 1 - p_c \cdot \frac{\delta(H)}{l-1}$$

która sprowadza się do poprzedniego oszacowania, gdy $p_c = 1$.

Przy założeniu niezależności operacji selekcji i krzyżowania średnia liczba reprezentantów danego schematu H w następnym pokoleniu po łącznym efekcie działania obu operatorów wynosi:

$$m(H,t+1) \geq m(H,t)\frac{f(H)}{\overline{f}}\left[1 - p_c \cdot \frac{\delta(H)}{l-1}\right]$$

Analiza powyższego wzoru pozwala wnioskować, że schematy, które mają przystosowanie wyższe od średniej i małą rozpiętość będą rozprzestrzeniać się zgodnie z wykładniczym prawem wzrostu.

Ostatnim elementem wchodzącym w skład teorii o schematach jest analiza wpływu na wzrost lub spadek reprezentacji schematów operatora mutowania. Operator ten zmienia zawartość każdej pozycji z prawdopodobieństwem równym pm. Aby schemat H przeżył mutację, muszą się zachować wszystkie jego pozycje ustalone. A

zatem, ponieważ pojedyncza pozycja chromosomu pozostaje bez zmian z prawdopodobieństwem $(1 - p_m)$ i ponieważ mutacje na poszczególnych pozycjach są statystycznie niezależne, dany schemat H przeżyje mutację z prawdopodobieństwem $(1 - p_m)o(H)$, gdzie o(H) jest liczbą pozycji ustalonych. Dla stosowanych w programach ewolucyjnych małych wartości p_m, prawdopodobieństwo przeżycia można aproksymować za pomocą wyrażenia $1 - o(H) \times p_m$. Można więc ostatecznie stwierdzić, że oczekiwana liczba reprezentantów schematu H w następnym pokoleniu, otrzymanym w wyniku reprodukcji, krzyżowania i mutacji, spełnia następującą nierówność:

$$m(H,t+1) \geq m(H,t)\frac{f(H)}{\overline{f}}\left[1 - p_c\frac{\delta(H)}{l-1} - o(H)p_m\right]$$

Wzór 1.11. pozwala na sformułowanie Twierdzenia o schematach albo Podstawowego twierdzenia algorytmów genetycznych: schematy dobrze przystosowane, niskiego rzędu i o małej rozpiętości rozprzestrzeniają się w kolejnych pokoleniach zgodnie z wykładniczym prawem wzrostu.

Jeśli szukają Państwo pomocy w napisaniu własnej pracy - potrzebują Państwo fachowych konsultacji to polecamy stronę pisanie prac - profesjonalna pomoc w pisaniu prac w granicach prawa.

Testowanie połączeń sieci LAN

Po podłączeniu wszystkich przewodów należy sprawdzić ciągłość połączeń. Do tego celu w najprostszym przypadku nadaje się multimetr z próbnikiem przejścia, lecz jest to uciążliwy proces wymagający uwagi i systematyki zwłaszcza, jeśli mamy do sprawdzenia więcej niż jedno połączenie.

Najlepiej nadaje się do tego tester przejścia do sieci (skrętkowych lub BNC, dzięki któremu można szybko się zorientować, czy kabel jest uszkodzony), lub urządzenie do wykonywania pomiarów sieci, które ponadto stwierdzi jakość okablowania.

Tester ciągłości połączeń, jest to proste urządzenie (nieco nowocześniejsza oraz ulepszona wersja bateryjki i żarówki) pozwalające wykryć:

- brak przewodzenia którejś z par skrętki
- kolejność podłączenia par skrętki
- prawidłowość polaryzacji każdej pary
- fakt zwarcia w kablu Tester do pomiarów sieci

Urządzenia do pomiarów sieci wykonują szereg testów (zazwyczaj automatycznie, po wciśnięciu jednego przycisku) i określają, czy dany parametr spełnia założenia danej normy (pass) lub nie (fail). Ocenie podlegają tu:

- Line Map – mapa połączeń;
- NEXT (Near End Crosstalk) – przesłuch pomiędzy parami;
- Return Loss – wartość sygnału odbitego będącego wynikiem niedopasowania impedancji elementów;
- Attenuation – tłumienie;
- Link Length – długość połączenia;

Niestety obecnie koszt testerów do wykonywania pomiarów w zależności od ich nowoczesności i uniwersalności sięga kilkuset do kilku tysięcy dolarów amerykańskich, więc jest to inwestycja, na którą mogą sobie pozwolić jedynie duże firmy.

estowanie połączeń sieci LAN (ang. Local Area Network) jest kluczowym elementem zarządzania infrastrukturą sieciową w każdej organizacji, niezależnie od jej wielkości. Współczesne systemy informatyczne opierają się na niezawodności i wydajności sieci lokalnych, które umożliwiają sprawną komunikację między komputerami, serwerami oraz innymi urządzeniami sieciowymi. Z tego względu, regularne testowanie

i monitorowanie połączeń sieci LAN stanowi niezbędny proces zapewnienia ich poprawnego działania, minimalizacji opóźnień oraz rozwiązywania potencjalnych problemów.

LAN to sieć obejmująca niewielki obszar, zazwyczaj ograniczony do jednego budynku lub kampusu, w której urządzenia komunikują się poprzez medium transmisyjne, takie jak kable Ethernet lub technologie bezprzewodowe. Ze względu na rosnące wymagania dotyczące przepustowości oraz niezawodności sieci, testowanie połączeń sieci LAN stało się bardziej skomplikowane i wymagające zaawansowanych narzędzi diagnostycznych oraz metod analizy.

Testowanie sieci LAN można podzielić na kilka etapów, które obejmują diagnozowanie różnych aspektów działania sieci. Jednym z podstawowych elementów testowania sieci LAN jest pomiar szybkości i jakości połączenia. Szybkość transmisji danych ma bezpośredni wpływ na wydajność aplikacji i użytkowników w sieci, dlatego testy przepustowości (ang. bandwidth test) są jednym z najważniejszych elementów oceny sprawności połączenia. Podczas tych testów mierzy się maksymalną ilość danych, jaka może zostać przesłana w jednostce czasu pomiędzy dwoma punktami w sieci. Ważnym aspektem tego procesu jest zarówno ocena przepustowości w kierunku wysyłania, jak i odbierania danych, co pozwala na uzyskanie pełnego obrazu wydajności sieci.

Kolejnym istotnym elementem testowania sieci LAN jest pomiar opóźnień (ang. latency). Opóźnienia są mierzone w czasie, jaki upływa pomiędzy wysłaniem zapytania z jednego urządzenia a otrzymaniem odpowiedzi od drugiego. Nawet jeśli sieć oferuje wysoką przepustowość, zbyt duże opóźnienia mogą znacząco obniżyć jej wydajność, zwłaszcza w przypadku aplikacji wrażliwych na czas reakcji, takich jak wideokonferencje czy gry online. W diagnostyce sieci LAN opóźnienia mogą być monitorowane za pomocą narzędzi takich jak ping czy traceroute, które pozwalają śledzić drogę pakietów przez sieć oraz identyfikować wąskie gardła lub problemy z połączeniem.

Kolejnym ważnym aspektem testowania sieci LAN jest ocena integralności połączenia i identyfikacja potencjalnych strat pakietów. Straty pakietów (ang. packet loss) mogą wystąpić w wyniku błędów transmisji, przeciążenia sieci lub problemów z infrastrukturą sprzętową, takich jak uszkodzone kable czy wadliwe urządzenia sieciowe. Wysoki poziom utraty pakietów może prowadzić do degradacji jakości usług, zwłaszcza w przypadku transmisji wideo i audio, które są szczególnie wrażliwe na utratę danych. Testy takie jak UDP (User Datagram Protocol) jitter czy TCP (Transmission Control Protocol) retransmissions są stosowane do monitorowania i identyfikacji problemów związanych z utratą pakietów w sieci LAN.

Testowanie stabilności połączenia w sieci LAN obejmuje także analizę zasięgu i jakości sygnału, zwłaszcza w przypadku sieci bezprzewodowych. W sieciach Wi-Fi, testowanie połączeń obejmuje mierzenie siły sygnału (ang. signal strength) oraz analizy interferencji, które mogą zakłócać stabilność połączenia. Interferencje w sieci bezprzewodowej mogą być spowodowane przez inne urządzenia działające na tych samych częstotliwościach, takie jak mikrofalówki czy inne sieci Wi-Fi w pobliżu. W rezultacie, monitorowanie jakości sygnału i minimalizacja zakłóceń jest kluczowa dla utrzymania stabilnych połączeń w sieciach bezprzewodowych.

W kontekście fizycznej warstwy sieci LAN, testowanie obejmuje także ocenę kabli i sprzętu sieciowego, takiego jak przełączniki czy routery. Przewody sieciowe mogą ulegać uszkodzeniom mechanicznym, co wpływa na jakość połączenia. Testowanie kabli Ethernet za pomocą narzędzi takich jak certyfikatory kabli pozwala na wykrycie problemów z okablowaniem, takich jak przerwy w obwodach, zwarcia czy niewłaściwe uziemienie. Testy te są szczególnie ważne podczas instalacji nowych sieci LAN, gdzie jakość kabli ma bezpośredni wpływ na stabilność i wydajność połączeń.

Współczesne testowanie sieci LAN nie ogranicza się jedynie do analizy fizycznej warstwy sieci, ale obejmuje również

monitorowanie protokołów sieciowych i ruchu sieciowego. Narzędzia takie jak sniffery sieciowe pozwalają na przechwytywanie i analizowanie pakietów danych przesyłanych w sieci, co umożliwia identyfikację problemów związanych z protokołami komunikacyjnymi. Analiza ruchu sieciowego pozwala także na wykrywanie nieautoryzowanych działań w sieci, takich jak ataki hakerskie czy nielegalne próby dostępu do zasobów.

W testowaniu połączeń sieci LAN coraz większą rolę odgrywają również technologie sztucznej inteligencji i uczenia maszynowego, które mogą automatycznie analizować duże zbiory danych sieciowych i identyfikować wzorce, które mogą sugerować problemy z połączeniami. Dzięki wykorzystaniu algorytmów analizy predykcyjnej, możliwe jest nie tylko wykrywanie bieżących problemów, ale także przewidywanie przyszłych awarii lub przeciążeń sieciowych, co umożliwia proaktywne zarządzanie infrastrukturą sieciową.

Warto również zauważyć, że w kontekście rosnącej popularności zdalnej pracy i nauki, testowanie połączeń sieci LAN stało się jeszcze ważniejsze. Zwiększone obciążenie sieci, wynikające z większej liczby urządzeń jednocześnie korzystających z zasobów sieciowych, stawia dodatkowe wymagania dotyczące przepustowości, stabilności i szybkości połączeń. Regularne testowanie i monitorowanie sieci LAN pozwala na szybkie reagowanie na zmiany w obciążeniu sieci oraz minimalizowanie przerw w dostępie do zasobów.

Testowanie połączeń sieci LAN jest procesem wieloetapowym, obejmującym zarówno analizę fizycznej infrastruktury, jak i monitorowanie wydajności, stabilności oraz bezpieczeństwa sieci. Współczesne narzędzia diagnostyczne pozwalają na szybkie wykrywanie i rozwiązywanie problemów, co przekłada się na zwiększenie niezawodności i efektywności sieci LAN w codziennym funkcjonowaniu organizacji. Testowanie sieci LAN nie tylko pomaga w utrzymaniu wysokiej jakości połączeń, ale także umożliwia przewidywanie przyszłych problemów i optymalizację zasobów sieciowych w celu sprostania rosnącym

wymaganiom użytkowników.

Jeśli szukają Państwo pomocy w napisaniu własnej pracy - potrzebują Państwo fachowych konsultacji to polecamy stronę pisanie prac - profesjonalna pomoc w pisaniu prac w granicach prawa.

Rozszerzenie funkcjonalności programu poprzez implementację reprezentacji przedziałowej

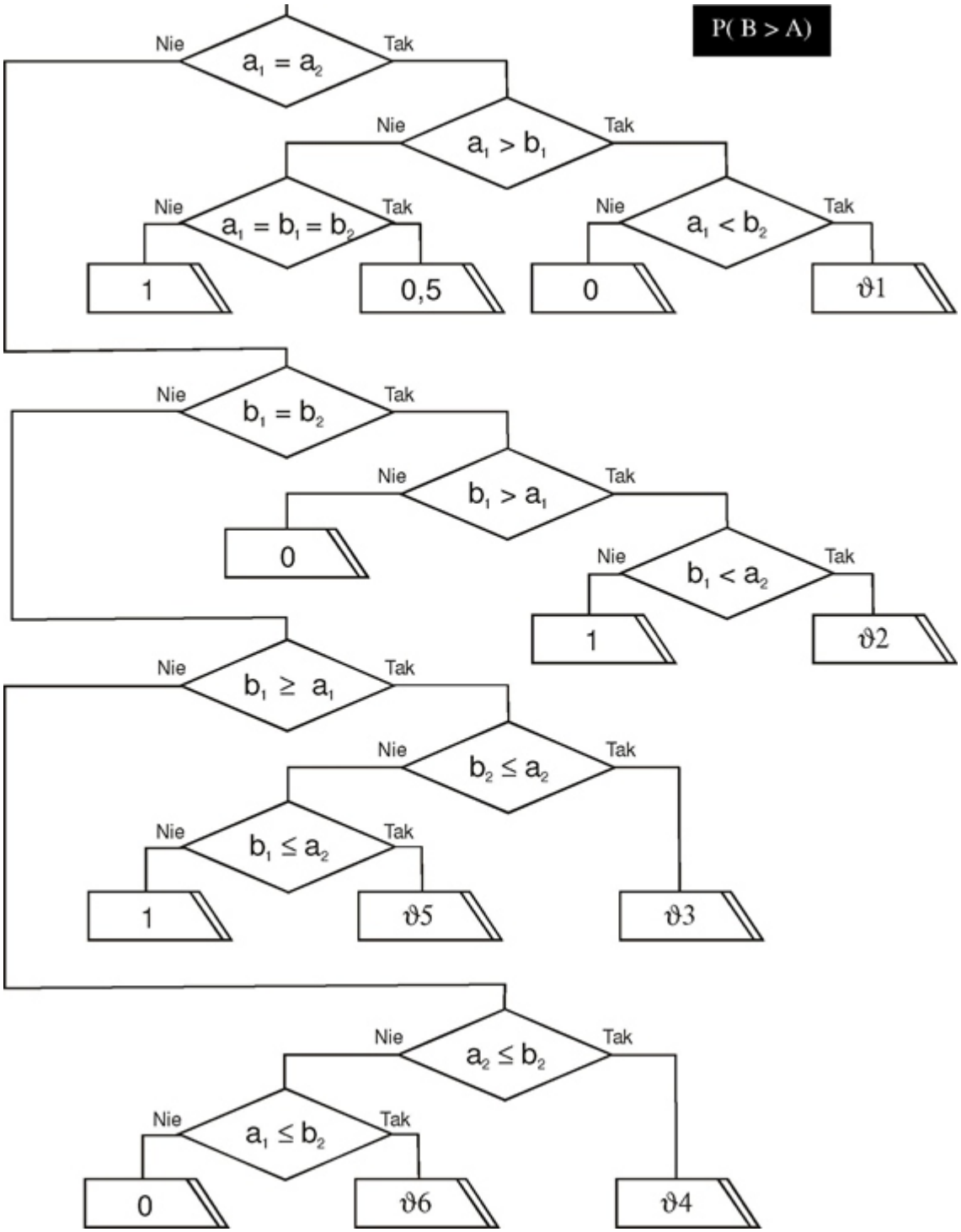
Praktyczne wykorzystanie programu GenMachine do optymalnej selekcji portfela papierów wartościowych, których oczekiwana stopa zwrotu reprezentowana jest przez przedziały ostre wymagało zaimplementowania dodatkowego obiektu C++ klasy Interval, w której znalazły się funkcje operatorowe realizujące działania arytmetyczne oraz porównywanie przedziałów.

Podane poniżej deklaracje metod klasy Interval pozwalaj ą na wykonywanie czterech podstawowych działań pomiędzy dwoma przedziałami oraz pomiędzy przedziałem i liczbą rzeczywistą. Ich implementacja oparta jest na tzw. “naiwnym” podejściu do problemu działań arytmetycznych, wyjaśnionym w rozdziale 3 [cytowanej pracy magisterskiej].

Interval operator += (const Interval &inter); // dodaj i przypisz Interval operator -= (const Interval &inter); // odejmij i przypisz Interval operator *= (const Interval &inter); // pomnóż i przypisz Interval operator /= (const Interval &inter); // podziel i przypisz

Interval operator += (real r); Interval operator -= (real r); Interval operator *= (real r); Interval operator /= (real r);

Ciekawym zagadnieniem okazała się implementacja metod porównywania wartości przedziałów na podstawie tabeli 3.1. Schemat blokowy na rysunku 4.2. pokazuje sposób porównania wartości przedziałów [a1 , a2] i [b1 , b2], obejmując wszystkie również trywialne przypadki względnego położenia przedziałów. Klatki wyjścia oznaczone symbolami od $1 do $6 wskazują na kolejne wzory w tabeli 3.1. Zaimplementowana w ten sposób funkcja operatorowa zwraca w wyniku liczbę rzeczywistą równą.



Rys. 4.2. Schemat blokowy operatora porównywania przedziałów [ai , a_2] i [bi , b_2]

Jeśli szukają Państwo pomocy w napisaniu własnej pracy - potrzebują Państwo fachowych konsultacji to polecamy stronę pisanie prac - profesjonalna pomoc w pisaniu prac w granicach prawa.

Interakcja MS z serwerem aplikacji

Interakcja serwera mediów z serwerem aplikacji jest kluczowa w kontekście realizacji usług multimedialnych w danej sieci V2OIP. Zagadnienie to wymaga zatem dokładnego rozważenia.

Poniżej opisano rolę, jaką pełni AS w interakcji z serwerem mediów. Dokonano prezentacji czterech zaproponowanych przez ETSI ([3GPP TR 24.880 "Media Server Control using the IP Multimedia (IM) Core Network (CN) subsystem", Maj 2007]) modeli sterowania serwerem mediów oraz przedstawiono krytyczne uwagi wskazując braki tych modeli. Zaproponowano także dodatkowy model oparty na wiadomości SIP REFER.

- Rola serwera aplikacji

Można wyróżnić trzy typy ról, jakie AS pełni wobec MS:

- Wywoływanie serwera mediów (invoke role) – serwer aplikacji wywołuje MS (wysyła żądania SIP) i w efekcie zestawia z nim ścieżkę sygnalizacyjną SIP.
- Sterowanie serwerem mediów (Control role) – AS decyduje, które spośród funkcjonalności serwera mediów zostaną wykorzystane i w jaki sposób
- Dostarczanie danych (service role) – AS udostępnia serwerowi mediów dane potrzebne do realizacji żądanych funkcji (np. adresy terminali, pliki a/v, skrypty VXML[1] etc) [2]

W szczególności:

- AS przeprowadza wszystkie operacje zgodnie z logiką zlokalizowanych w nim aplikacji usługowych

- AS korzysta z B2BUA[89] [3] celem zestawienia połączeń (sesji RTP) pomiędzy MS a terminalami SIP, sterowania przebiegiem i parametrami tych połączeń
- AS wykorzystuje funkcjonalności multimedialne MS poprzez protokół SIP oraz (w sytuacji gdy SIP nie wspiera odpowiedniej funkcjonalności) poprzez dedykowany protokół sterowania serwerem mediów, którego wiadomości są transportowane w ciele wiadomości protokołu SIP.
- AS może odbierać od MS następujące informacje:

o Stan pracy MS (zajęte zasoby, pojemność MS etc.) o Stan realizacji żądanych funkcjonalności (zakończone sukcesem, niepowodzeniem, przerwane)

o Dane na temat realizowanych funkcjonalności (np. dla konferencji – lista uczestników, ich adresy, kodeki, lista N najgłośniejszych uczestników etc).

o Zaistniałe błędy i awarie w pracy MS o Pobrane od użytkowników cyfry i symbole DTMF o Rezultaty rozpoznawania mowy użytkownika (np. dla menu głosowego)

Informacje te są przesyłane poprzez wiadomości SIP lub wiadomości dedykowanego protokołu na styku AS-MS

- W kontekście usługi konferencji AS:

o Implementuje funkcjonalność bloku:

- Focus[4] [5]
- Conference Policy Server

o Jest wywoływane przez abonentów poprzez adres konferencji (Conference URI)[6]

o Jest wykonawcą polityk konferencji – cyklu życiowego konferencji (Life Cycle), zarządzania członkami konferencji (Membership), autoryzacji (Authorisation), za wyjątkiem tzw. polityki dotyczącej mediów (media conferencepolicy, [41]),

która jest wykonywana przez serwer mediów (MRFC lub warstwę sterowania w MS). o Może implementować Conference Notification Server, poprzez wykorzystanie mechanizmu opartego na wiadomościach SIP SUBSCRIBE oraz NOTIFY, jak opisano w IETF RFC 4575 "A Session Initiation Protocol (SIP) Event Package for Conference State", Sierpień 2006.

- W kontekście funkcjonalności billingowej AS[7] [8]:

o Implementuje różne modele billingowe związane z konferencją: prepaid, postpaid, dzielenie opłaty pomiędzy uczestników, opłata za konferencję, opłata w zależności od wykorzystywanego kodeka etc. o Wspiera naliczanie opłaty za innego typu funkcjonalności: opłata za odtwarzanie pliku audio/video, opłata za ilość przesłanych danych, za jakość kodeka/szerokość wykorzystywanego pasma, opłata za transkodowanie etc.

Sprzężenie AS-MS

Często można spotkać stwierdzenie, iż serwer mediów to moduł, który samodzielnie odpowiada za realizację usługi multimedialne. W taki sposób np. reklamowane są rzeczywiste implementacje MS, a ich producenci często przytaczają zestaw usług, które są wg nich realizowane przez ich produkt.

Należy zwrócić uwagę na fakt, iż jest to stwierdzenie błędne. Jak wspomniano w rozdziale 5, MS nie zawiera żadnej logiki usługowej, a jest jedynie pewnego rodzaju „biernym procesorem sygnałowym", który realizuje żądania innych modułów sieciowych.

Przyjmijmy zatem następującą tezę:

W sieciach V2OIP zachodzi sprzężenie funkcjonalne serwera aplikacji (AS) i serwera mediów (MS), w którym to AS steruje pracą serwera mediów i wykorzystuje jego zasoby zgodnie z logiką aplikacji usługowej, a MS udostępnia funkcjonalności przetwarzania strumieni multimedialnych. Z tego powodu w

kontekście realizacji usług multimedialnych należy zawsze rozpatrywać współpracę obu modułów.

W odniesieniu do przytoczonej w rozdziale 4 definicji usługi multimedialnej, można powiedzieć, że to MS dostarcza niezbędnych funkcjonalności, natomiast AS decyduje o sposobie i kolejności ich wykorzystania.

Poparciem powyżej tezy niech będzie fakt, iż ETSI w dokumencie [41] definiuje moduł Conferencing Application Server (Konferencyjny Serwer Aplikacji), który odpowiada za logikę tej właśnie usługi i jest „nieodzownie" sprzężony z serwerem mediów [9] (oba moduły tworzą „blok konferencyjny").

---

[1] Patrz rozdział 6.6.4.1. [cytowanej **pracy magisterskiej**]

[2] Przy rejestrowaniu SIP UA odpowiadających funkcjonalnościom Media Servera nie ujętym w notacji Netann, niestandardowe nazewnictwo w SIP URI jest konieczne. Notacja wspiera bowiem tylko trzy grupy funkcjonalności (patrz rozdział 6.6.2.3).

[3] Funkcja B2BUA została omówiona w rozdziale 4 niniejszej pracy.

[4] IETF RFC 4353 „A Framework for Conferencing with the Session Initiation Protocol (SIP)", Luty 2006, patrz też rozdział 2.2.

[5] Jest to logiczny blok funkcjonalny, który przechowuje polityki konferencji (conference policies)i udostępnia je blokowi focus do zaadoptowania. Blok ten może być np. zintegrowany z focusem. Opisano go w IETF RFC 4353 „A Framework for Conferencing with the Session Initiation Protocol (SIP)".

[6] IETF RFC 4353 „A Framework for Conferencing with the Session Initiation Protocol (SIP)”, patrz też rozdział 2.2. [cytowanej **pracy magisterskiej**]

[7] Jest to logiczny blok, będący częścią focusa. Opisano go w IETF RFC 4353 „A Framework for Conferencing with the Session Initiation Protocol (SIP)”. Umożliwia przesyłanie powiadomień o stanie konferencji tym użytkownikom, którzy dokonali subskrypcji związanej z tym stanem.(mechanizmy powiadamiania i subskrypcji z wykorzystaniem protokołu SIP opisano w IETF RFC 3265 “Session Initiation Protocol (SIP)-Specific Event Notification”, Czerwiec 2002).

[8] Dokładna analiza roli serwera aplikacji w realizacji bililingu leży poza zakresem niniejszej pracy.

[9] Informacje na temat rozproszenia pomiędzy serwer aplikacji i Media Server (MRF) bloków funkcjonalnych dla usługi konferencji opartej na protokole SIP, zdefiniowanych w RFC 4353, zawarto w 3GPP TR 24.880 “Media Server Control using the IP Multimedia (IM) Core Network (CN) subsystem”, Maj 2007.

Jeśli szukają Państwo pomocy w napisaniu własnej pracy - potrzebują Państwo fachowych konsultacji to polecamy stronę pisanie prac - profesjonalna pomoc w pisaniu prac w granicach prawa.

XSL

praca dyplomowa inżynierska

Język XSL (Extensible Stylesheet Language) stanowi uzupełnienie XML. W założeniu ma on służyć do formatowania

dokumentów w XML (podobnie jak CSS dla HTML). Może być on również używany do przekształcania dokumentów XML na dokumenty HTML. XSL nie tylko umożliwia definiowanie wyglądu, ale także selekcj ę, dokonywanie operacji oraz sortowanie danych. XSL definiuje szablon końcowy oraz określa w jaki sposób elementy źródłowe mają być przekształcone i w jaki sposób umieszczone w dokumencie HTML. Aktualnie ogólnodostępne przeglądarki nie potrafią interpretować opisu formatowania wyrażonego w XSL. Za to doskonale radzą sobie z prezentacją dokumentów HTML. Zatem dokumenty XSL przetwarzane zostają do postaci HTML.

Wg najnowszego podziału, wyróżnia się cztery części języka:

- XSLT (XSL Transformations) – zawiera zestaw instrukcji umożliwiających przekształcenie dokumentu XML'owego w nowy dokument,
- XPATH – zawiera język do adresowania części dokumentu XML.
- XSL (Extensible Stylesheet Language) – dawny XSL-FO, słownik opisujący formatowanie,
- XQuery (XML Query Language) – język zapytań.

> *„Prezentacja dokumentu z użyciem XSL polega na transformacji drzewa hierarchii znaczników dokumentu wejściowego na drzewo tzw. formatting objects, którym przypisano określone sposoby prezentacji. Przeglądarka służąca do prezentacji powinna zrozumieć język owych obiektów formatujących."*[18]

Proces transformacji dokumentu:

1. parsowanie oryginalnego dokumentu XML'owego,
2. parsowanie dokumentu XSLT – poprawny dokument XML'a, zawiera opis transformacji jaką należy wykonać na oryginalnym dokumencie XML'a,
3. utworzenie wynikowego dokumentu XML, który zawiera znaczniki z XMLF,
4. interpretacja znaczników opisujących sposób formatowania

i prezentacji dokumentów.

Dla XSLT istnieje wiele implementacji kompilatorów, które pozwalają na praktyczne skorzystanie z możliwości jakie oferuje XML+XSLT. Transformacje opisywane przez XSLT są zapisywane jako poprawny dokument XML, który może zawierać zarówno elementy należące do specyfikacji języka XSL, jak i elementy użytkownika. Transformacja wyrażona w XSLT zawiera zbiór reguł opisujących przekształcenie jednego drzewa XML w nowe. Reguły te wyrażane są przy pomocy wzorców, które składają się z części pozwalającej na sprawdzenie czy dana reguła powinna zostać zastosowana do bieżącego elementu drzewa XML i z zawartości wzorca czyli tego co ma się pojawić w wynikowym drzewie po napotkaniu pasującego elementu drzewa. Elementy dokumentu XML są kolejno dopasowywane do wzorców i w ten sposób, poruszając się w głąb źródłowego drzewa, tworzone jest drzewo dokumentu wynikowego.

XSLFO jest skomplikowanym językiem o dużych możliwościach, zawierającym ponad 50 różnych „obiektów formatujących”. Obiekty te można formatować wykorzystując ponad 200 różnych właściwości, takich jak: kroje, odmiany i wielkości pisma, odstępy, kolory itp.

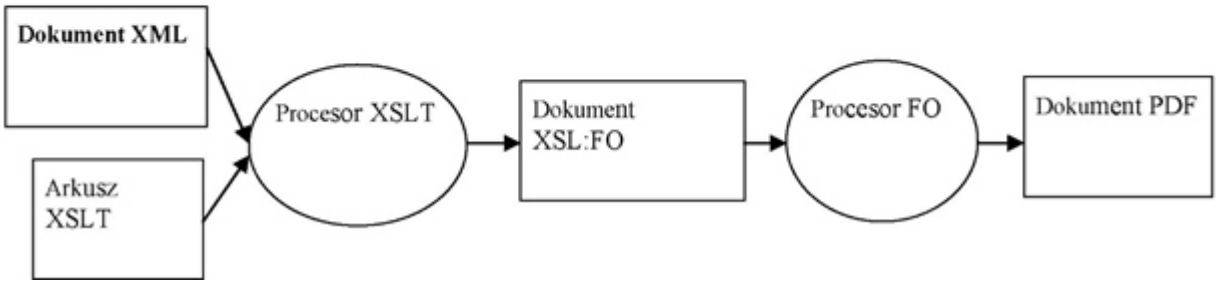
W dokumencie XSLFO elementy i atrybuty nie opisują struktury dokumentu, a wyłącznie jego postać graficzną. Różni go to od arkuszy stylów CSS, w których polecenia formatujące są (zwykle) oddzielone od dokumentu, którego dotyczą. XSLFO pod tym względem przypomina raczej tradycyjne systemy przetwarzania tekstu takie jak TeX. Dokument XSLFO winien być generowany automatycznie na podstawie danych zapisanych w innym formacie (np. „strukturalnego” dokumentu XML lub informacji z relacyjnej bazy danych), a po wykorzystaniu usunięty. Format XSLFO nie jest przeznaczony ani do przechowywania informacji, ani do jej wymiany.

W typowym scenariuszu dokument XSLFO jest tworzony za pomocą odpowiedniego arkusza stylu XSLT. W poniższym przykładowym

fragmencie arkusza XSLT element para wyjściowego dokumentu jest transformowany do elementu fo:block:

Rysunek 11. Typowy scenariusz przetwarzania dokumentu XSLFO

```
<template match="para">
 <fo:block space-after.optimum="0pt" hyphenate="true" >
<xsl:apply-templates/> </fo:block>
</xsl:template>
```



Cały dokument XSLFO zawarty jest wewnątrz elementufo:root, który składa się z:

- jeden elementfo:layout-master-set zawierający szablony określające wygląd poszczególnych stron oraz sekwencji stron,
- zero lub więcej elementów fo:declarations,
- jeden lub więcej elementówfo:page-sequance zawierających treść formatowanego dokumentu wraz z opisem jego sformatowania i podziału na strony.

Treść formatowanego dokumentu znajduje się wyłącznie wewnątrz elementów fo:page-sequence, podczas gdy elementfo:layout-master-set zawiera definicje szablonów.

---

[18] „JęzykXSL”, Tomasz Traczyk, IAilS, Politechnika Warszawska, ia.pw.edu.pl/~ttraczyk/pdf/ploug2000_art.pdf

Jeśli szukają Państwo pomocy w napisaniu własnej pracy - potrzebują Państwo fachowych konsultacji to polecamy stronę pisanie prac - profesjonalna pomoc w pisaniu prac w granicach prawa.

Protokoły komunikacyjne

Protokół jest zbiorem reguł normujących transmisję danych oraz formaty przesyłanych komunikatów. W celu wymiany danych, komputery podłączone do sieci muszą korzystać z protokołów komunikacyjnych. Przesył danych w sieci od bywa się w formie komunikatów zdefiniowanych przez protokół komunikacyjny. Opisuje on również sposoby postępowania z komunikatami zarówno w wypadku wystąpienia błędu, jak również wtedy, kiedy transmisja przebiega poprawnie. Wszystkie programy korzystające z Internetu współpracują z odpowiednimi protokołami.

SGML jest międzynarodowym standardem (ISO 8879:1986) definiującym metody reprezentacji tekstu w postaci elektronicznej, niezależnej od platformy sprzętowej i systemu operacyjnego.

Za twórców języka SGML uważa się Charlesa Goldfarba, Edwarda Moshera oraz Raymonda Loriego, którzy w 1969 roku opracowali dla IBM-u język GML (Generalized Markup Language – również akronim nazwisk twórców). GML był rezultatem projektu zintegrowanego systemu informacyjnego dla kancelarii prawniczych i pozwalał na edytowanie, formatowanie oraz przeszukiwanie danych tekstowych. Autorzy projektu w znacznej mierze opierali się na wcześniejszych propozycjach Tunnicliffe’a i Rice’a. W wyniku kontynuowania prac przez Goldfarba nad rozwojem GML, w 1980r. powstała pierwsza robocza wersja standardu SGML, a w 1986r. ukazała się międzynarodowa wersja robocza i przyjęto SGML, jako standard międzynarodowy.

SGML jest metajęzykiem, który określa zasady tworzenia języków opisujących klasy dokumentów. Takim językiem, opracowanym na bazie SGML, jest popularny w Internecie HTML (HyperText Markup

Language). SGML pozwala na oznaczanie tekstu w sposób ogólny (ang. generic markup), zwany również oznaczaniem opisowym (ang. descriptive markup). Umieszczone w tekście znaczniki niosą informację znaczeniową i w przeciwieństwie do oznaczania proceduralnego (ang. procedural markup), nie informują systemu o wyglądzie dokumentu, rodzaju użytych czcionek, rozmieszczeniu elementów tekstu na stronie. SGML opisuje nie tylko zawartość dokumentu (tekst, zdjęcia, animacje), lecz również jego strukturę logiczną (słowo, akapit, rozdział, sekcja). Pozwala także na tworzenie dokumentów składających się z wielu plików, zachowując jego integralność. Obiektem opisującym tekst są znaczniki. Na początku i na końcu każdego z nich znajdują się znaki (ograniczniki), które pozwalają na wyraźne oddzielenie obiektów opisujących tekst od zasadniczej treści dokumentu.

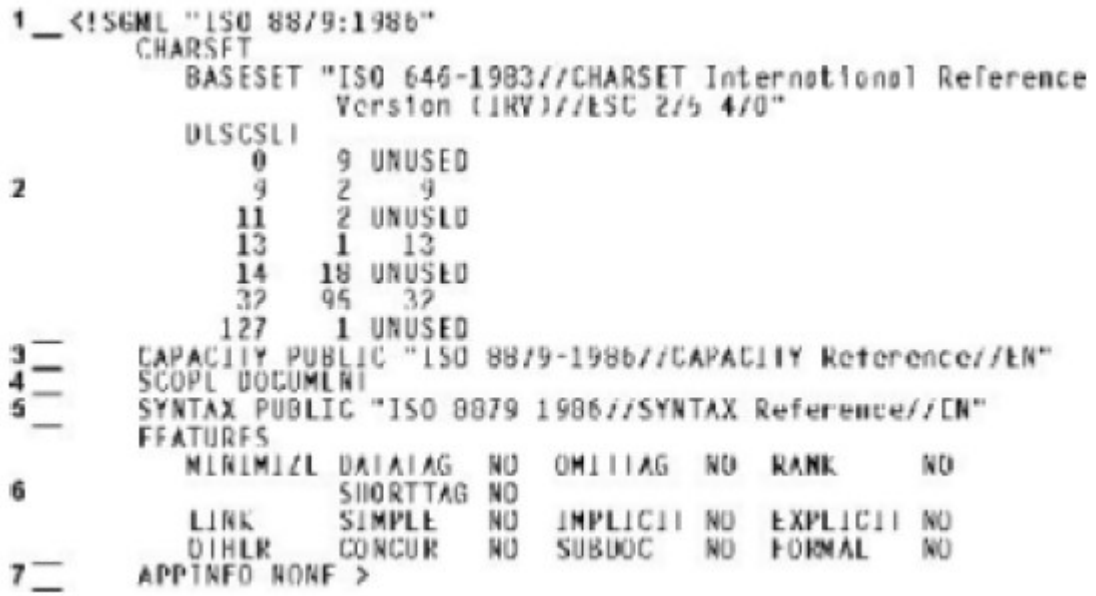
Dokument zgodny z SGML składa się z dwóch podstawowych części:

- prolog SGML (ang. SGML prolog),
- dokument zasadniczy (ang. document instance).

Prolog zawiera deklarację SGML (ang. SGML declaration) oraz definicję typu dokumentu (ang. Document Type Definition – DTD). Dokument zasadniczy zawiera tekst, który jest oznaczony zgodnie z DTD umieszczonym w prologu.

Deklaracja SGML zawiera podstawowe dane o strukturze leksykalnej używanej w DTD oraz w dokumencie zasadniczym. Definiuje m.in. dozwolony zestaw znaków, maksymalną liczbę znaków w nazwach (elementów, atrybutów lub encji), maksymalną liczbę zadeklarowanych elementów w DTD, dozwolone ograniczniki (ang. delimiters). Definiuje również wykorzystanie cech dopuszczanych przez SGML, takich jak opuszczanie znaczników (OMITTAG) oraz skracanie znaczników (SHORTTAG). W wielu przypadkach deklaracja SGML może zostać opuszczona. Oznaczać to będzie, że do DTD oraz dokumentu zasadniczego zastosowane zostaną reguły zdefiniowane w Reference Concrete Syntax, dostarczane przez większość systemów SGML. Jeżeli jednak

będziemy chcieli dokonać jakichkolwiek zmian (nowy zestaw znaków, polskie znaki diakrytyczne w nazwach, rozróżnianie małych i dużych liter w nazwach), wówczas deklaracja SGML musi pojawić się w całości. Przykładowa deklaracja SGML przedstawia się następująco:



Rysunek 10 Deklaracja SGML

1. Opisowy nagłówek
2. Zestaw znaków
3. Zestaw pojemności
4. Zakres Składni Konkretnej (ang. Concrete Syntax Scope).
5. Składnia Konkretna (ang. Concrete Syntax).
6. Użycie dodatkowych cech (ang. Features Use).
7. Informacje dla aplikacji (ang. Application-Specific Information).

Definicja Typu Dokumentu (Document Type Definition; DTD) jest jednym z podstawowych pojęć związanych z SGML-em. DTD jest to sformalizowany opis struktury dokumentów pewnego typu, zawierający spis ich elementów składowych oraz hierarchię elementów właściwą dla danego typu dokumentu. Można powiedzieć, że dopiero DTD jest definicją języka (czy formatu) do opisywania struktury danej klasy dokumentów.

Do oznaczania elementów w dokumentach służą znaczniki (tags).

Każdemu elementowi przypisany jest znacznik otwierający (nazwa) i znacznik zamykający (/nazwa). Oprócz definicji elementów i znaczników z nimi związanych DTD zawiera także reguły ich używania: np. czy możliwe jest opuszczenie znacznika zamykającego lub otwierającego dany element, jeśli z kontekstu możliwe jest wywnioskowanie faktu jego istnienia.

DTD tworzy się zgodnie z prostą składnią zdefiniowaną w standardzie. Jest to zbiór deklaracji, który rozpoczyna się deklaracj ą typu dokumentu DOCTYPE.



DTD może być również przedstawione w postaci systemowego lub publicznego identyfikatora (ang. system, public identifier). Identyfikator jest częścią deklaracji DOCTYPE i pozwala zlokalizować zasoby systemowi SGML przetwarzającemu dokumenty.

Identyfikator systemowy (ang. system identifier) odsyła do źródła w lokalnych zasobach:

<!DOCTYPE book SYSTEM ”book.dtd”>

<!DOCTYPE manual SYSTEM ”C:\sgml\dtd\manual.dtd”>

Identyfikator publiczny (ang. public identifier) jest najczęściej przedstawiany w postaci sformalizowanej jako formalny publiczny identyfikator (ang. formal public identifier). W tym przypadku tworzymy odsyłacz do tekstu publicznego (ang. public text). Tekstem publicznym może być DTD, jego fragment, zbiór encji, zestaw znaków, a nawet cały dokument SGML udostępniany do powszechnego użycia.

Formalny publiczny identyfikator składa się z dwóch zasadniczych części:

identyfikatora właściciela (ang. owner identifier),

identyfikatora tekstu (ang. text identifier).

Przykłady DTD z formalnym publicznym identyfikatorem:

<!DOCTYPE book PUBLIC ”-//USA/AAP//DTD BK-1//EN”>

<!DOCTYPE sd PUBLIC ”+//British Library//DTD Starter Document//EN”>

Znaki – oraz + wskazują, czy identyfikator właściciela został zarejestrowany.

Mamy wówczas pewność, że nie istniej ą różne DTD o tych samych nazwach. Do rejestracji uprawniona jest Registration Authority ustanowiona przez International Organisation for Standarisation (ISO). Przydziela ona nazwę, która powinna pojawić się na początku identyfikatora właściciela. Procedura rejestracyjna została dokładnie opisana w standardzie ISO 9070.[11]

Jeśli szukają Państwo pomocy w napisaniu własnej pracy - potrzebują Państwo fachowych konsultacji to polecamy stronę pisanie prac - profesjonalna pomoc w pisaniu prac w granicach prawa.

Protokoły sieciowe

praca inżynierska z informatyki w zarządzaniu

Protokół sieciowy wyjaśnia cały uprzednio uzgodniony przez nadawcę i odbiorcę proces wymiany danych na określonej warstwie modelu sieciowego. W uproszczonym czterowarstwowym modelu sieciowym można wyróżnić następujące protokoły:

- Protokoły warstwy fizycznej: Ethernet, Token Ring
- Protokoły warstwy sieciowej: protokół internetowy (IP) będący częścią zestawu protokołów TCP/IP
- Protokoły warstwy transportu: protokół sterowania transmisją w sieci (TCP/IP) i protokół datagramów użytkownika (UDP), które są częścią zestawu protokołów TCP/IP
- Protokoły warstwy aplikacyjnej: protokół przesyłania plików (FTP), prosty protokół przysłania poczty (SMTP), usługi nazewnicze domen (DNS), telnet, protokół przesyłania hipertekstu (HTTP), prosty protokół zarządzania siecią (SNMP), które także są częścią zestawu protokołów TCP/IP

Określenie „zestaw protokołów" oznacza dwa lub więcej protokołów z tych warstw, które stanowią podstawę sieci.

Kilka najlepiej znanych zestawów protokołów to: zestaw protokołów IPX/SPX („międzysieciowa wymiana pakietów"/,, sekwencyjna wymiana pakietów") używany przez system Novell Netware, NetBIOS i NetBEUI („rozszerzony interfejs użytkownika podstawowego sieciowego systemu wejścia/wyjścia") zaprojektowane przez firmę IBM, wykorzystywany m.in. przez system operacyjny Microsoftu. Ponadto NetBIOS może być tunelowany dowolnym innym protokołem np. IPX lub TCP/IP – zestaw protokołów TCP/IP („protokół kontroli transmisji'7„protokół internetowy") używany powszechnie w Internecie oraz sieciach lokalnych mających do niego dostęp.

Protokoły sieciowe są podstawą działania współczesnych sieci komputerowych, umożliwiając komunikację między urządzeniami w sieci. Dzięki nim możliwe jest przesyłanie danych, zarządzanie połączeniami oraz udostępnianie zasobów na całym świecie. Protokoły definiują zasady, według których urządzenia wymieniają informacje, dzięki czemu mogą rozumieć wysyłane i odbierane komunikaty. W świecie technologii sieciowej, protokoły te zapewniają, że niezależnie od rodzaju urządzenia czy technologii, dane są wymieniane w sposób spójny i

efektywny.

Protokoły sieciowe to zestawy reguł i standardów, które kontrolują sposób, w jaki dane są transmitowane i odbierane w sieci komputerowej. Istnieje wiele różnych rodzajów protokołów sieciowych, z których każdy ma swoje specyficzne funkcje i zastosowania, odpowiadające różnym warstwom modelu OSI (ang. Open Systems Interconnection), który jest referencyjnym modelem dla standardów komunikacyjnych w sieciach komputerowych. Model OSI dzieli komunikację sieciową na siedem warstw, a każda z nich odpowiada za inny aspekt transmisji danych. W ramach tego modelu protokoły są przypisane do różnych warstw, umożliwiając niezależne zarządzanie poszczególnymi funkcjami.

Kluczowe protokoły sieciowe w modelu OSI

Model OSI to struktura odniesienia, która opisuje, jak dane są przesyłane w sieci, od warstwy fizycznej, przez warstwę aplikacji, po użytkownika końcowego. Każda warstwa pełni specyficzną funkcję, a komunikacja między nimi odbywa się poprzez określone protokoły. W ramach tego modelu, niektóre z najbardziej kluczowych protokołów obejmują:

1. **Warstwa fizyczna (Layer 1)**: Na tym poziomie definiowane są fizyczne aspekty transmisji, takie jak okablowanie, sygnały elektryczne, czy fale radiowe w przypadku sieci bezprzewodowych. Choć protokoły bezpośrednio nie funkcjonują na tej warstwie, standardy takie jak Ethernet (IEEE 802.3) są fundamentem komunikacji.
2. **Warstwa łącza danych (Layer 2)**: W tej warstwie protokoły zajmują się bezpośrednią komunikacją między sąsiadującymi urządzeniami w sieci lokalnej (LAN). Przykładowymi protokołami na tym poziomie są Ethernet oraz protokół Point-to-Point (PPP). Warstwa ta zapewnia, że dane są przesyłane między urządzeniami bez błędów, umożliwiając komunikację na poziomie ramki danych (ang. data frame).

3. **Warstwa sieciowa (Layer 3)**: Jednym z najważniejszych protokołów na tej warstwie jest **IP (ang. Internet Protocol)**, który zarządza adresowaniem i routowaniem pakietów danych w sieci. IP umożliwia identyfikację każdego urządzenia w sieci poprzez unikalne adresy IP, co pozwala na precyzyjne dostarczenie danych z jednego miejsca do drugiego. Protokół IP dzieli się na dwie główne wersje: IPv4 i IPv6, z których ta druga, dzięki większej liczbie dostępnych adresów, staje się coraz bardziej popularna w miarę wzrostu liczby urządzeń podłączonych do internetu.
4. **Warstwa transportowa (Layer 4)**: Na tym poziomie najbardziej powszechne protokoły to **TCP (Transmission Control Protocol)** oraz **UDP (User Datagram Protocol)**. TCP jest protokołem zapewniającym niezawodność transmisji danych poprzez mechanizm potwierdzeń oraz retransmisji utraconych pakietów. Używany jest tam, gdzie kluczowe jest dokładne dostarczenie danych, na przykład w przesyłaniu plików czy przeglądaniu stron internetowych. Z kolei UDP jest protokołem bardziej lekkim, który nie zapewnia niezawodności, ale za to oferuje szybszą transmisję, dlatego jest stosowany tam, gdzie kluczowa jest prędkość, na przykład w transmisjach wideo na żywo.
5. **Warstwa sesji (Layer 5)**: Warstwa sesji zarządza ustanawianiem, utrzymaniem oraz zamykaniem połączeń pomiędzy aplikacjami na różnych urządzeniach. Przykładami protokołów na tym poziomie są **NetBIOS** oraz **RPC (Remote Procedure Call)**, które zapewniają zarządzanie sesjami i synchronizację komunikacji.
6. **Warstwa prezentacji (Layer 6)**: Na tej warstwie dane są konwertowane do formatu, który może być przetworzony przez aplikacje wyższego poziomu. Protokoły na tej warstwie zajmują się również szyfrowaniem i kompresją danych. Jednym z przykładów protokołów funkcjonujących na tej warstwie jest **SSL/TLS** (ang. Secure Sockets Layer / Transport Layer Security), który zapewnia bezpieczne połączenia internetowe.

7. **Warstwa aplikacji (Layer 7)**: Najbardziej widoczna dla użytkownika warstwa, obejmująca protokoły umożliwiające komunikację między aplikacjami na różnych urządzeniach. Przykładami są **HTTP/HTTPS** (ang. Hypertext Transfer Protocol / Hypertext Transfer Protocol Secure), które są fundamentem działania przeglądarek internetowych, **FTP (ang. File Transfer Protocol)** do przesyłania plików oraz **SMTP (ang. Simple Mail Transfer Protocol)**, który jest używany do przesyłania wiadomości e-mail.

Protokoły internetowe i ich znaczenie

Najbardziej znane i powszechnie stosowane protokoły sieciowe to te, które są związane z internetem, czyli globalną siecią komputerową. **IP** jest najważniejszym z nich, ponieważ odpowiada za trasowanie i przesyłanie danych pomiędzy różnymi sieciami. Protokół IP działa w połączeniu z innymi protokołami, które tworzą kompleksowy system zapewniający sprawną komunikację w internecie.

TCP, wspomniany wcześniej, jest kluczowym protokołem transportowym. Zapewnia on niezawodność przesyłania danych dzięki mechanizmom potwierdzeń, retransmisji oraz kontrolowania przepływu danych, co sprawia, że jest szeroko stosowany w takich aplikacjach, jak przeglądanie stron internetowych czy przesyłanie plików. Jego lżejszym odpowiednikiem jest **UDP**, który, dzięki swojej prostocie, znajduje zastosowanie tam, gdzie najważniejsza jest szybkość transmisji, a ewentualne straty danych są akceptowalne, np. w streamingu multimediów czy grach online.

W kontekście współczesnego internetu, coraz ważniejszą rolę odgrywa również **DNS (Domain Name System)**, który przekształca adresy internetowe (np. www.example.com) na adresy IP, zrozumiałe dla komputerów. DNS to jeden z kluczowych protokołów ułatwiających korzystanie z sieci WWW, pozwalający użytkownikom na łatwe wyszukiwanie stron internetowych za pomocą nazw domenowych, a nie trudnych do zapamiętania adresów

IP.

Innym istotnym protokołem związanym z bezpieczeństwem internetowym jest **HTTPS**, który jest bezpieczną wersją protokołu HTTP. Dzięki wykorzystaniu szyfrowania SSL/TLS, HTTPS zapewnia poufność i integralność danych przesyłanych pomiędzy serwerem a przeglądarką internetową, co ma szczególne znaczenie w przypadku transakcji finansowych czy przesyłania danych osobowych.

Wyzwania i przyszłość protokołów sieciowych

Z biegiem czasu i wraz z rozwojem technologii sieciowych, rosną wymagania wobec protokołów, które muszą sprostać wyzwaniom związanym z rosnącym ruchem sieciowym, większymi wymaganiami odnośnie bezpieczeństwa oraz dynamicznie rozwijającą się infrastrukturą. Jednym z największych wyzwań jest przejście z IPv4 na IPv6, które jest odpowiedzią na problem wyczerpujących się adresów IP. IPv6 oferuje znacznie większą pulę dostępnych adresów, co jest kluczowe w kontekście rozwoju Internetu Rzeczy (IoT) oraz coraz większej liczby urządzeń podłączonych do sieci.

Innym wyzwaniem jest zapewnienie bezpieczeństwa w dynamicznie zmieniającym się środowisku sieciowym. W miarę wzrostu liczby cyberzagrożeń, protokoły muszą ewoluować, aby sprostać rosnącym wymaganiom dotyczącym ochrony danych i integralności połączeń. Protokoły takie jak TLS są stale aktualizowane, aby oferować wyższy poziom zabezpieczeń przed atakami.

W przyszłości możemy spodziewać się jeszcze większej elastyczności i wydajności protokołów sieciowych, które będą musiały dostosować się do rosnących wymagań wynikających z globalnego rozwoju sieci 5G, Internetu Rzeczy oraz technologii opartych na sztucznej inteligencji.

Protokoły sieciowe są podstawą nowoczesnej komunikacji

cyfrowej, umożliwiając przesyłanie danych między urządzeniami w sposób zrozumiały i efektywny. Ich złożona struktura, oparta na modelu OSI, pozwala na niezawodne i bezpieczne funkcjonowanie sieci, od lokalnych sieci LAN po globalny internet. Współczesne protokoły, takie jak IP, TCP, UDP czy DNS, odgrywają kluczową rolę w codziennym funkcjonowaniu internetu, zapewniając sprawną wymianę informacji między miliardami urządzeń na całym świecie. W miarę rozwoju technologii sieciowych, rola protokołów będzie nadal rosła, a ich ewolucja stanie się fundamentem dla przyszłych innowacji w dziedzinie komunikacji i transmisji danych.

Jeśli szukają Państwo pomocy w napisaniu własnej pracy - potrzebują Państwo fachowych konsultacji to polecamy stronę pisanie prac - profesjonalna pomoc w pisaniu prac w granicach prawa.

Urządzenia aktywne LAN

podrozdział piąty pracy magisterskiej

Sieci LAN buduje się z biernych i aktywnych urządzeń sieciowych. Bierne urządzenia sieciowe to komponenty systemów okablowania strukturalnego.

Do aktywnych urządzeń sieci LAN należą:

- regenerator (repeater) – jest urządzeniem pracującym w warstwie fizycznej modelu OSI, stosowanym do łączenia segmentów kabla sieciowego. Regenerator odbierając sygnały z jednego segmentu sieci wzmacnia je, poprawia ich parametry czasowe i przesyła do innego segmentu.

Może łączyć segmenty sieci o różnych mediach transmisyjnych.

- koncentrator (hub) – jest czasami określany jako wieloportowy regenerator. Służy do tworzenia fizycznej gwiazdy przy istnieniu logicznej struktury szyny lub pierścienia. Pracuje w warstwie 1 (fizycznej) modelu OSI. Pakiety wchodzące przez jeden port są transmitowane na wszystkie inne porty.

Wynikiem tego jest fakt, że koncentratory pracują w trybie half-duplex (transmisja tylko w jedną stronę w tym samym czasie).

- przełącznik (switch) – są urządzeniami warstwy łącza danych (warstwy 2) i łączą wiele fizycznych segmentów LAN w jedną większą sieć. Przełączniki działają podobnie do koncentratorów z tą różnicą, że transmisja pakietów nie odbywa się z jednego wejścia na wszystkie wyjścia przełącznika, ale na podstawie adresów MAC kart sieciowych przełącznik uczy się, a następnie kieruje pakiety tylko do konkretnego odbiorcy, co powoduje wydatne zmniejszenie ruchu w sieci. W przeciwieństwie do koncentratorów, przełączniki działają w trybie full-duplex (jednoczesna transmisja w obu kierunkach).

Przełączniki działają w oparciu o jeden z dwóch trybów pracy: cut through (przełączanie bezzwłoczne) oraz store&forward (zapamiętaj i wyślij). Pierwsza technologia jest wydajniejsza, ponieważ pakiet jest natychmiast kierowany do portu przeznaczenia (na podstawie MAC adresu) bez oczekiwania na koniec ramki, lecz pakiety przesyłane w taki sposób nie są sprawdzane pod względem poprawności. Druga technologia pracy charakteryzuje się tym, że przełącznik odczytuje najpierw całą ramkę, sprawdza, czy została odczytana bez błędów i dopiero potem kieruje ją do portu docelowego. Przełącznik taki pracuje wolniej, ale za to prawie niezawodnie.

- przełącznik VLAN – jest odmianą przełącznika

umożliwiającą tworzenie wirtualnych sieci LAN, których stanowiska są zlokalizowane w różnych punktach (sieciach, podsieciach, segmentach), zaś w sieć wirtualną łączy je jedynie pewien klucz logiczny. Sieć taka pozwala optymalizować natężenie ruchu pakietów w poszczególnych częściach sieci. Możliwa jest również łatwa zmiana konfiguracji oraz struktury logicznej takiej sieci.

- most (bridge) – służy do przesyłania i ew. filtrowania ramek między dwoma sieciami przy czym sieci te niekoniecznie muszą być zbudowane w oparciu o takie samo medium transmisyjne. Śledzi on adresy MAC umieszczane w przesyłanych do nich pakietach. Mosty nie mają dostępu do adresów warstwy sieciowej, dlatego nie można ich użyć do dzielenia sieci opartej na protokole TCP/IP na dwie podsieci IP. To zadanie mogą wykonywać wyłącznie routery. Analizując adresy sprzętowe MAC, urządzenie wie, czy dany pakiet należy wyekspediować na drugą stronę mostu, czy pozostawić bez odpowiedzi.

Mosty podobnie jak przełączniki przyczyniają się w znacznym stopniu do zmniejszenia ruchu w sieci.

- router – urządzenie wyposażone najczęściej w kilka interfejsów sieciowych LAN, porty obsługujące sieć WAN, pracujący wydajnie procesor i oprogramowanie zawiadujące ruchem pakietów przepływających przez router. W sieciach lokalnych stosowane są, gdy sieć chcemy podzielić na dwie lub więcej podsieci. Segmentacja sieci powoduje, że poszczególne podsieci są od siebie odseparowane i pakiety nie przenikają z jednej podsieci do drugiej. W ten sposób zwiększamy przepustowość każdej podsieci.

Jeśli szukają Państwo pomocy w napisaniu własnej pracy - potrzebują Państwo fachowych konsultacji to polecamy stronę pisanie prac - profesjonalna pomoc w pisaniu prac w granicach prawa.

Konstruowanie populacji genotypów w postaci ciągów binarnych

Program ewolucyjny oparty na klasycznym algorytmie genetycznym operuje na zbiorze, którego liczba elementów jest stała, a każdy z elementów jest dopuszczalnym rozwiązaniem zadania. Zgodnie z terminologią zapożyczoną z genetyki naturalnej elementy zbioru możliwych rozwiązań nazywane są najczęściej osobnikami, genotypami lub chromosomami, natomiast ich zbiór nazywany jest populacją.

Pierwszym krokiem, przed rozpoczęciem głównego procesu symulacji ewolucji genotypów, jest ustalenie sposobu reprezentacji rozwiązania zadania takiego, aby przy jego pomocy można było przedstawić każdą permutację należących do dziedziny wartości zmiennych decyzyjnych. W klasycznej wersji algorytmu genetycznego używa się chromosomu w postaci wektora binarnego. Taki właśnie sposób kodowania użyty zostanie w omawianym dalej przykładzie optymalizacji funkcji jednej zmiennej.

Długość ciągu binarnego reprezentującego rzeczywistą zmienną decyzyjną zależy od długości dziedziny funkcji i żądanej dokładności obliczeń. Przy założeniu, że wymagana jest dokładność do 6 cyfr po przecinku oraz że dziedzina zmienności x ma długość 3 i zawiera się w przedziale $[-1, 2]$, przedział ten musi zostać podzielony na $3 \cdot 1.000.000$ równych podprzedziałów. Oznacza to, że genotyp musi mieć 22 bity, gdyż

$$2.097.152 = 2^{21} < 3.000.000 \leq 2^{22} = 4.194.304$$

Odwzorowanie łańcucha binarnego < b21 b20 … b0 > w liczbę rzeczywistą x z przyjętego zakresu [– 1, 2] można wykonać w dwóch krokach: przekształcenie łańcucha binarnego < b21 b20 … b0 > z systemu dwójkowego na dziesiętny:

$$(<b_{21}b_{20}..b_0>)_2 = \left(\sum_{i=0}^{21} b_i 2^i\right)_{10} = x'$$

(1.1.)

obliczenie odpowiedniej liczby rzeczywistej x: x = -1,0 + 3x’ / (2^{22}-1), gdzie -1 jest lewą granicą dziedziny, a 3 jest długością przedziału.

Przy takim sposobie kodowania genotypy (0000000000000000000000) i (1111111111111111111111) reprezentują odpowiednio granice dziedziny, – 1,0 i 2,0.

W tym rozdziale omówiony zostanie prosty przykład optymalizacji funkcji f(x) = x² przy pomocy odręcznej symulacji klasycznego algorytmu genetycznego. W przykładzie tym, zmienna decyzyjna x może przybierać tylko wartości całkowite z przedziału [0 , 31] , a zatem do reprezentacji całej dziedziny wystarczy kod pięciobitowy, którego minimalną wartością jest 0 (00000) natomiast maksymalną 31 (11111). Odwzorowanie takiego łańcucha < b4 b3 … b0 > w liczbę całkowitą, wymaga tylko jednego kroku – bezpośredniej zamiany liczby binarnej na dziesiętną.

Po wyborze sposobu reprezentacji genotypu przychodzi czas na losowy wybór populacji początkowej, czyli z reguły od 50 do 100 ciągów binarnych tej samej długości. W omawianym przykładzie populacja składa się z czterech osobników, pierwsze pokolenie można zatem utworzyć przy pomocy 20 rzutów rzetelną monetą.

Tabela 1.1. Populacja początkowa wygenerowana losowo

Numer genotypu	Populacja początkowa (wygenerowana losowo)	Wartość x (liczba całkowita)
1	01101	13
2	11000	24
3	01000	8
4	10011	19

Konstruowanie funkcji celu i funkcji przystosowania genotypów Podstawowym elementem każdego programu ewolucyjnego jest reprodukcja, czyli proces, w którym indywidualne ciągi kodowe zostają powielone w stosunku zależnym od wartości, jakie przybiera dla nich funkcja celu. Funkcja celu jest zatem kluczowym elementem programu, treścią zadania do rozwiązania, środowiskiem w jakim żyją osobniki. Najczęściej jest to funkcja wielu zmiennych, o nieznanej przeciwdziedzinie, bardzo trudna lub niemożliwa do rozwiązania metodą analizy matematycznej. Funkcja celu stanowi pewien miernik zysku, użyteczności lub innej wartości, którą należy zmaksymalizowć.

W klasycznym wydaniu algorytmu genetycznego prawdopodobieństwo wyboru osobnika do reprodukcji jest wprost proporcjonalne do jego użyteczności. Tymczasem wiele zadań wyraża się w bardziej naturalny sposób w kategoriach minimalizacji pewnej funkcji kosztu niż maksymalizacji pewnej funkcji użyteczności lub zysku. Nawet, jeśli zadanie w naturalny sposób dotyczy maksymalizacji, nie gwarantuje to, że funkcja użyteczności będzie przyjmowała wartości wyłącznie nieujemne. Wskutek tego często zachodzi potrzeba przekształcenia oryginalnej funkcji celu w funkcję przystosowania (nieujemne kryterium jakości), czego można dokonać w jednym lub kilku krokach.

W badaniach operacyjnych, przejście od zadania minimalizacji do zadania maksymalizacji, polega na pomnożeniu funkcji kosztu przez minus jeden. W przypadku algorytmu genetycznego taka operacja jest niewystarczająca, gdyż otrzymany miernik nie

musi być zawsze nieujemny. Stosuje się zamiast tego powszechnie następujące przekształcenie kosztu g(x) w przystosowanie f(x):

$$f(x) = \begin{cases} C_{max} - g(x), & \text{jeżeli } g(x) < C_{max} \\ 0 & \text{w przeciwnym przypadku} \end{cases} \quad (1.2.)$$

Współczynnik Cmax może być równy największej napotkanej do tej pory wartości funkcji g(x), albo największej wartości g(x) w populacji bieżącej lub w kilku ostatnich populacjach. Może też wahać się w zależności od wariancji populacji.

Kiedy w zadaniu występuje w naturalny sposób funkcja zysku lub użyteczności i nie wykonuje się przejścia z zadania minimalizacji do maksymalizacji, ujemne wartości funkcji celu nadal sprawiają trudności. Nadal konieczne jest przekształcenie funkcji celu u(x) w funkcję przystosowania f(x):

$$f(x) = \begin{cases} u(x) + C_{min}, & \text{jeżeli } u(x) + C_{min} > 0 \\ 0 & \text{w przeciwnym przypadku} \end{cases}$$

Współczynnik Cmin jest najczęściej wartością bezwzględną najmniejszej wartości u(x) w populacji bieżącej lub kilku ostatnich populacjach, może też być funkcją wariancji dostosowania populacji.

Omawiany przykład polega na maksymalizacji funkcji f(x) = x2 , gdzie x ∈ [0, 31]. Tak sformułowana funkcja celu nie wymaga przekształcenia aby zostać funkcją przystosowania. Przyjmuje wyłącznie wartości nieujemne, zatem w niezmienionej formie umożliwia zastosowanie reprodukcji różnicującej szansę wyboru osobników proporcjonalnie do ich przystosowania. Wybrane wcześniej losowo genotypy mogą już zostać ocenione, a co za tym idzie może zostać oszacowana szansa każdego z nich na przetrwanie i reprodukcję:

Tabela 1.2. Ocena pierwszego pokolenia

Numer genotypu	Ciąg binarny	Wartość x	f(x) 2 x	Prawdopodobieństwo wyboru	Oczekiwana liczba kopii
1	01101	13	169	0,14	0,58
2	11000	24	576	0,49	1,97
3	01000	8	64	0,06	0,22
4	10011	19	361	0,31	1,23
Suma:			1170	1,00	4,00
Średnia:			293	0,25	1,00
Maksimum:			576	0,49	1,97

Jeśli szukają Państwo pomocy w napisaniu własnej pracy - potrzebują Państwo fachowych konsultacji to polecamy stronę pisanie prac - profesjonalna pomoc w pisaniu prac w granicach prawa.