

Optymalizacja sieci FTTH

Hoang Nghia Le

Fiber To The Home (FTTH) to najbardziej nowoczesna technologia wśród światłowodowych sieci dostępowych. Podstawowymi wyzwaniami w procesie budowy sieci FTTH są wysoka cena urządzeń i koszt instalacji kabli. Cena urządzeń optycznych wprawdzie stale się obniża, jednak położenie kabla optycznego nadal generuje bardzo duże koszty. W niniejszym artykule przedstawiono problem optymalizacji kosztów instalacji kabla optycznego. Jego rozwiązanie będzie miało praktyczne zastosowanie w projektowaniu sieci FTTH. Ponieważ prezentowany problem jest NP-trudny (Non-deterministic Polynomial-time hard), w celu jego rozwiązania posłużono się algorytmem aproksymacyjnym. Algorytm został zaimplementowany w celu analizy wydajności, która wykazała zadowalającą jakość wyników działania algorytmu z dopuszczalną czasochłonnością. W związku z tym autor proponuje zastosowanie algorytmu w narzędziach do projektowania sieci FTTH.

FTTH, Fiber To The Home, sieć telekomunikacyjna, sieć optyczna, optymalizacja kosztów, instalacja kabla optycznego

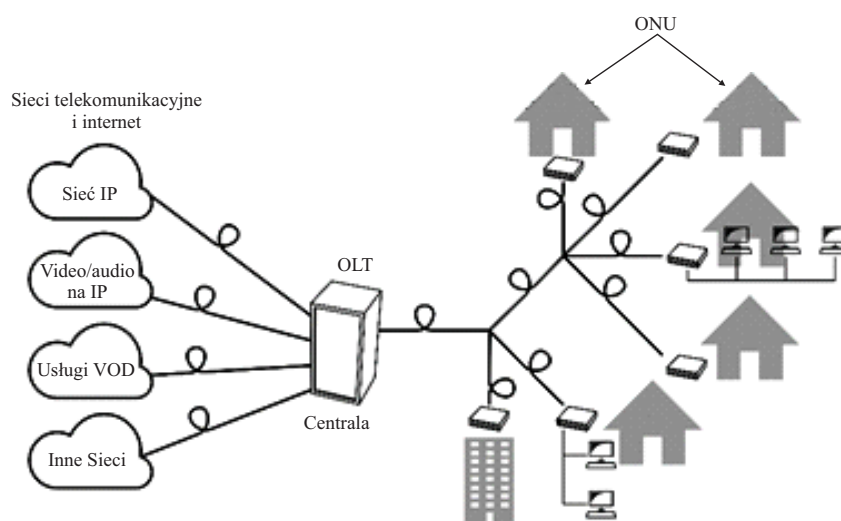
Wprowadzenie

Fiber To The Home (FTTH) to najbardziej nowoczesne rozwiązanie wśród światłowodowych sieci dostępowych. W przeciwieństwie do innych technologii strumień danych cyfrowych jest w niej przesyłany przez kabel optyczny bezpośrednio do terminala odbiorcy. Dzięki temu FTTH umożliwia transmisję danych z większą wyższą szybkością i lepszą jakością, niż w innych sieciach (radiowych, miedzianych, koncentrycznych lub optyczno-koncentrycznych). FTTH jest technologią, której zalety w coraz większym stopniu będą się ujawniać w niedalekiej przyszłości. Ilość przesyłanych danych rośnie bowiem wykładniczo. W przypadku sieci FTTH transmisja większej ilości danych wymaga jedynie instalacji szybszych terminali i routerów, przepustowość kabla pozostanie przez długi czas wystarczająca.

Mimo ewidentnych zalet tempo rozwoju technologii FTTH na świecie jest bardzo nierówne. W rocznym rankingu sporządzonym przez FTTH Council na temat stopnia pokrycia siecią FTTH (odsetek użytkowników Internetu korzystających z sieci FTTH) w poszczególnych krajach [1] wskazuje się grupę około 20 krajów, na czele z Koreą Południową, Zjednoczonymi Emiratami Arabskimi i Japonią, z rozwiniętą technologią FTTH (stopień pokrycia powyżej 25%). W pozostałych krajach odsetek ten nie przekracza 5%. Barięą dla rozwoju technologii FTTH są wysokie koszty instalacji urządzeń i kabli światłowodowych.

Jak widać na rys. 1, sieć FTTH zaczyna się w urządzeniu dystrybucyjnym *Optical Line Terminal* (OLT), który jest punktem styku z siecią szkieletową. Sygnały optyczne, które przenoszą strumienie danych cyfrowych do różnych użytkowników sieci, na początku są transmitowane we wspólnym

kablu światłowodowym. Jest to możliwe dzięki technologii *Wave Division Multiplexing* (WDM), która umożliwia jednoczesną transmisję kilkudziesięciu kanałów optycznych w jednym włóknie światłowodowym, a każdy z tych sygnałów wykorzystuje oddzielne pasmo. W kolejnym etapie po rozdzieleniu przez rozgałęźnik (splitter) sygnał jest przesyłany oddzielnymi kablami i dociera do odbiorników zlokalizowanych w domach użytkowników sieci *Optical Network Units* (ONU). W sieci FTTH nie są wykorzystywane kable miedziane, jak ma to miejsce w sieciach innych technologii. Dzięki temu sygnałów nie trzeba konwertować z optycznych na elektryczne. Konwersja taka poważnie obniża maksymalną szybkość transmisji danych dla pojedynczego użytkownika.



Rys. 1. Struktura sieci FTTH

Cena urządzeń sieciowych (na rys. 1 OLT i ONU) systematycznie spada, podczas gdy koszt instalacji kabli pozostaje na tym samym poziomie stając się główną przeszkodą w rozwoju sieci FTTH na świecie. Koszt ten składa się z kilku części składowych:

- kopania rowów telekomunikacyjnych,
- położenia rur telekomunikacyjnych w istniejących rowach,
- położenia kabli w rurach,
- kabla.

Niniejszy artykuł prezentuje nową metodę optymalizacji sieci FTTH koncentrującą się na minimalizacji kosztu instalacji kabli. Po analizie dotychczasowych prac na temat optymalizacji sieci FTTH zagadnienie optymalizacji przedstawione zostało w postaci problemu matematycznego, który nazwany został problemem *FTTH Network Optimization* (FNO). Autor skonstruował algorytm dokładny dla FNO, modelując go jako problem programowania liniowego całkowitoliczbowego *Integer Linear Programming* (ILP). Ponieważ problem FNO jest NP-trudny, algorytm dokładny pozwala na znalezienie rozwiązania tylko dla małych instancji FNO (dla mniej niż dziesięciu użytkowników końcowych). Dla większych instancji FNO zaproponowano algorytm aproksymacyjny, który za cenę rezygnacji z poszukiwania rozwiązania idealnego, pozwala na wyznaczenie rozwiązania bliskiego optimum, ograniczając jednocześnie złożoność obliczeniową i umożliwiając zastosowanie algorytmu w systemach komputerowych o cenie akceptowalnej dla projektantów sieci. W celu oceny algorytmów wyniki ich działania zaprezentowano na przykładach.

2. Powiązane prace naukowe

Zagadnienia optymalizacji w planowaniu sieci FTTH były przedmiotem licznych badań naukowych [2], które koncentrowały się na następujących obszarach:

- optymalizacja sposobu instalacji urządzeń [3],
- optymalizacja sposobu instalacji kabli [4]–[7],
- optymalizacja wykorzystania przepustowości w istniejących sieciach FTTH [8].

Przedstawiony w [3] optymalizowany koszt obejmuje dwie części: CAPEX (koszt instalacji urządzeń) i OPEX (koszt utrzymania elementów sieci). Problem optymalizacji sieci przekształcono do postaci problemu programowania liniowo-całkowitoliczbowego ILP. W związku z tym, że problem jest NP-trudny, zaproponowano rozwiązanie aproksymacyjne. Optymalizacja sposobu instalacji kabli nie jest w [3] wzięta pod uwagę.

W [4]–[7] wzięto wprawdzie pod uwagę zagadnienie optymalizacji sposobu instalacji kabli, jednak autorzy ograniczyli zakres badania do problemu wyboru najlepszej spośród predefiniowanych konfiguracji położenia kabli. Podejście takie czyni wynik optymalizacji mocno zależnym od arbitralnie zdefiniowanej konfiguracji położenia kabli. W [7] wzięte jest pod uwagę dodatkowo zagadnienie niezawodności sieci – po awarii pojedynczego kabla sieć powinna działać bez strat.

W [8] rozważana jest inna kategoria optymalizacji sieci optycznej – optymalizacja sposobu dystrybucji przepustowości w istniejącej sieci FTTH.

Po analizie wyników dotychczasowych badań autor doszedł do wniosku, że problem optymalizacji sieci FTTH w pełnej formie optymalizacji geometrycznej – znalezienie optymalnej alokacji urządzeń i kabli na danym obszarze przy uwzględnieniu uwarunkowań geograficznych – nie został rozwiązany. Uzupełnienie tej luki było celem niniejszej pracy badawczej.

Optymalizacja sieci z uwzględnieniem uwarunkowania geograficznego ma zastosowanie w wielu dziedzinach: energetyce, transporcie, przemyśle, a także w telekomunikacji (w szczególności dla sieci szkieletowych). Podstawowymi problemami optymalizacji tego typu są problem komiwojażera *Traveling Salesman Problem* (TSP) oraz problem drzewa Steinera w przestrzeni euklidesowej *Euclidean Steiner Tree* (EST) [9]–[10]. Pierwszy z tych problemów polega na znalezieniu najkrótszej trasy łączącej zbiór punktów w przestrzeni euklidesowej. Drugi polega na znalezieniu najmniejszej sieci łączącej zbiór punktów na płaszczyźnie.

Autor przeanalizował szereg prób rozszerzania EST o aspektach geometrycznych przeprowadzonych w dotychczasowych pracach [11]–[12]. W tych pracach autorzy proponują algorytm znalezienia drzewa Steinera, które unika przeszkód terenowych. W tym opracowaniu zagadnienie to zostało rozszerzone o następujące aspekty dodatkowe:

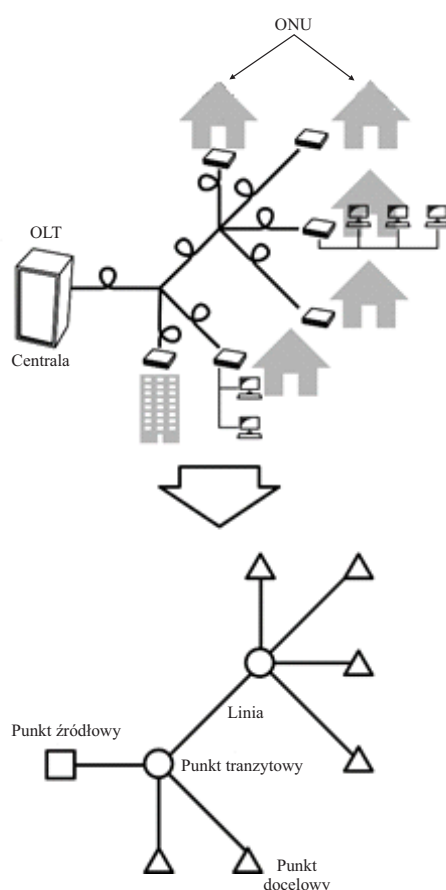
- Aspekt geograficzny. W problemie EST celem optymalizacji jest znalezienie najkrótszej sieci. W praktyce taka sieć nie zawsze jest najtańsza, jeśli niektóre odcinki muszą przechodzić przez teren z utrudnieniem lub brakiem możliwości instalacji kabli. W tej pracy koszt instalacji kabli wynikający z uwarunkowania terenowego jest przypisany do każdego punktu na płaszczyźnie, zaś algorytm znajduje taką sieć, która ma najmniejszy sumaryczny koszt;
- Ograniczoną przepustowość kabla;
- Istniejące zasoby sieci. Jest to istotne rozszerzenie, które pozwala na użycie algorytmu w szerokim zastosowaniu praktycznym, w częstych przypadkach, kiedy projektant ma do czynienia z rozbudową istniejącej sieci.

3. Problem optymalizacji sieci FTTH

W niniejszej sekcji przedstawiono formalny zapis problemu optymalizacji sieci *FTTH Network Optimization* (FNO). Dane wejściowe geometryczne są następujące:

- dyskretny i skończony zbiór punktów źródłowych na płaszczyźnie $S \subset R^2$ (punkt źródłowy reprezentuje urządzenie OLT),
- dyskretny i skończony zbiór punktów docelowych na płaszczyźnie $D \subset R^2$ (punkt docelowy reprezentuje urządzenie ONU),
- dyskretny i skończony zbiór istniejących punktów tranzytowych na płaszczyźnie $T1 \subset R^2$ (punkt tranzytowy reprezentuje rozgałęźnik),
- dyskretny i skończony zbiór linii łączących punkty w zestawy S, D i $T1$: $E1 \subset (S \cup D \cup T1)$ (linia reprezentuje odcinek kabla).

Na rys. 2 przedstawiono model przykładowej sieci FTTH.



Rys. 2. Model sieci FTTH

Dane wejściowe ilościowe są następujące:

- przepustowość kabla C - oznacza maksymalną liczbę kanałów WDM, które mogą być przesyłane przez kabel;

- funkcja gęstości kosztu instalacji kabla w obszarze $g: R^2 \rightarrow R$. Koszt instalacji kabla wynika z ukształtowania terenu oraz istniejącej na nim infrastruktury. Koszt ten może być: bardzo niski, jeśli kabel ma być położony wzdłuż drogi, dość wysoki, jeśli kabel ma być położony w poprzek drogi lub nieskończony, jeśli kabel ma być położony w poprzek budynku lub osiedla (instalacja jest niemożliwa).

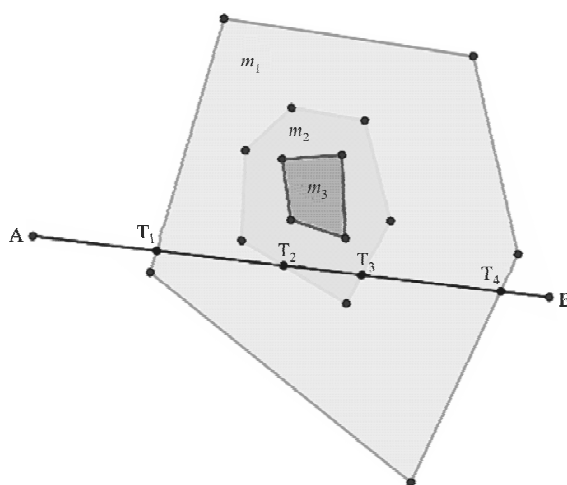
Funkcja $g(x)$ opisuje koszt instalacji kabla o jednostkowej długości w punkcie $x \in R^2$. Znając funkcję g można wyliczyć koszt instalacji linii e za pomocą:

$$f(e) = \int_{r=0}^l g((1-r) \cdot \text{in}(e) + r \cdot \text{out}(e)) dr \quad (1)$$

gdzie $\text{in}(e)$ oznacza punkt wejściowy linii e , $\text{out}(e)$ oznacza punkt wyjściowy linii e .

Założono, że funkcja g jest opisana za pomocą skończonego i posortowanego zbioru wielokątów, z których każdy ma przypisaną wartość kosztu. Wielokąty te nazywamy g -wielokątami. Wartość $g(x)$ dla punktu x w R^2 to koszt pierwszego wielokąta, który zawiera x . Jeśli żaden z wielokątów nie zawiera x , to $g(x) = 1$. G -wielokąt reprezentuje obszar o takim samym koszcie instalacji kabla, na przykład obszar wzdłuż drogi (niski koszt), obszar w poprzek drogi (wysoki koszt), obszar pod budynkiem (nieskończenie wysoki koszt).

Na rys. 3 koszt instalacji kabla z A do B to $f(AB) = \|AT_1\| + \|T_1T_2\| \cdot m_1 + \|T_2T_3\| \cdot m_2 + \|T_3T_4\| \cdot m_1 + \|T_4B\|$, gdzie m_1, m_2, m_3 są odpowiednio koszty oznaczonych tymi symbolami obszarów.



Rys. 3. Koszt położenia kabla na obszarze g -wielokątów

Zapis problemu optymalizacji jest następujący:

Znajdź:

- skończony zbiór punktów tranzytowych $T \subset R^2$,
- skończony zbiór linii łączących punkty w zbiorze S, D i T : $E \subset (S \cup D \cup T)^2$,
- skończony zbiór ścieżek (sekwencji połączonych linii) $P = \{p: p = (e_1, e_2, \dots, e_n); \forall i = 1..n, e_i \in E, \forall j = 1..n-1, \text{out}(e_j) = \text{in}(e_{j+1})\}$.

Warunki:

- warunek łączności – każdy punkt docelowy jest połączony z przynajmniej jednym punktem źródłowym, albo bezpośrednio jedną linią, albo poprzez sekwencję połączonych linii. Innymi słowy, dla każdego punktu $d \in D$ istnieje ścieżka $p \in P$ taka, że $\text{in}(p) \in S$ oraz $\text{out}(p) = d$, gdzie $\text{in}(p)$ oznacza punkt wejściowy pierwszej linii ścieżki p a $\text{out}(p)$ oznacza punkt wyjściowy ostatniej linii ścieżki p ;
- **warunek przepustowości** – liczba ścieżek przechodzących przez linię nie powinna przekroczyć predefiniowanej przepustowości linii:

$$\forall e \in E, \sum_{p \in P: e \in p} 1 \leq C,$$

gdzie w kontekście sieci z technologią WDM, C jest maksymalną liczbą kanałów, które mogą być transmitowane przez jeden kabel optyczny.

Cel optymalizacji:

Celem optymalizacji jest znalezienie rozwiązania z najmniejszym sumarycznym kosztem instalacji kabli:

$$\text{zminimalizować } \sum_{e \in E} f(e) \quad . \quad (2)$$

4. Algorytm dokładny do FNO

W niniejszej pracy zaproponowano dwa algorytmy do rozwiązania problemu FNO. Główny to algorytm aproksymacyjny, który umożliwia skuteczne odnalezienie prawie optymalnego rozwiązania problemu. Drugi z zaproponowanych algorytmów to algorytm dokładny przekształcony do postaci programowania liniowego całkowitoliczbowego (ILP). Algorytm ten jest skuteczny tylko dla małych instancji problemu i na potrzeby niniejszego opracowania jest wykorzystany jedynie do oceny jakości wyniku działania głównego algorytmu.

Problem FNO jest silnie związany z rodziną problemów drzewa Steinera, w szczególności dwóch z nich:

- topologiczny problem drzewa Steinera (*Topological Steiner Tree*, TST) – mając podany graf zawierający zbiory węzłów V i łączy E (każde łącze ma ustaloną wartość kosztu), połączyć podany podzbiór T zbioru V za pomocą podgrafu z najmniejszą sumą kosztu łączy.
- problem drzewa Steinera w przestrzeni euklidesowej (*The Euclidean Steiner Tree*, EST) – mając podane n punktów na płaszczyźnie (zwanymi punktami końcowymi), połączyć je za pomocą odcinków z najmniejszą sumaryczną długością w taki sposób, że każda para punktów końcowych może być połączona odcinkami albo bezpośrednio, albo poprzez inne punkty (zwane punktami Steinera).

TST będzie zastosowany wielokrotnie jako procedura w algorytmach dla FNO, natomiast EST jest specjalnym przypadkiem FNO, gdy S jest zbiorem składającym się z jednego elementu, g jest funkcją stałą i C jest większy niż $|D|$.

Ponieważ EST był intensywnie studiowany w ubiegłym wieku, został skonstruowany algorytm dokładny bazujący na wiedzy zebranej w pracach na temat EST. Udowodniono, że EST jest problemem klasy NP-trudny [13]. Trudność w transformowaniu EST i jego pochodnych problemów (takich jak FNO) do postaci programowania liniowo-całkowitoliczbowego (ILP) polega na tym, że zbiór podanych punktów do optymalizacji jest zbiorem ciągłym i nieskończonym, podczas gdy postać ILP powinna zawierać dyskretne i skończone zbiory zmiennych decyzyjnych i równań.

W celu transformowania FNO do postaci ILP, trzeba więc „dyskretyzować” przestrzeń danych tego problemu. Innymi słowy konieczne jest skonstruowanie dyskretnego i skończonego zbioru punktów, który zawiera szukane punkty Steinera, tzw. punkty kandydujące.

4.1. Punkty kandydujące dla EST i FNO

Istnieją dwa podejścia dla dyskretyzacji przestrzeni problemu EST i FNO:

- poprzez wprowadzenie siatki na płaszczyźnie, gdzie wierzchołki siatki stanowią arbitralne punkty kandydujące,
- poprzez generowanie punktów kandydujących przy wykorzystaniu właściwości geometrycznej problemu EST.

Pierwsze podejście jest dość praktyczne dla algorytmów aproksymacyjnych, jednak nie pozwala na znalezienie rozwiązania dokładnego. Jakość wyniku działania algorytmu zależy od ziarnistości siatki. Dlatego w proponowanym algorytmie dokładnym zastosowano drugie podejście.

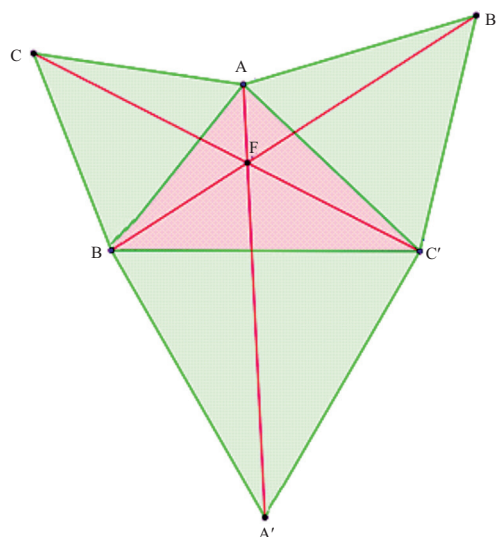
Najpierw wyszukiwany jest zbiór punktów kandydujących dla EST, a potem ten zbiór jest rozszerzany dla FNO. Aby oznaczyć problem EST dla podanej wartości n , zastosowano symbol $EST(n)$.

Przypadek $EST(2)$ jest trywialny. Rozwiązaniem problemu jest odcinek prosty łączący podane dwa punkty. $EST(3)$ jest równoważny do problemu znalezienia punktu Fermata – mając podany trójkąt ΔABC , znajdź taki punkt F na płaszczyźnie, że suma odległości od niego do wierzchołków trójkąta jest najmniejsza.

Problem ten został po raz pierwszy ogłoszony w 1643 roku przez słynnego francuskiego matematyka Fermata, jako wyzwanie dla włoskiego matematyka Torricellego.

Fermat rozwiązał problem (rys. 4) poprzez:

- skonstruowanie trzech równobocznych trójkątów ($\Delta BCA'$, $\Delta ACB'$ i $\Delta ABC'$), każdy z nich korzysta z jednego z boków podanego trójkąta,
- skonstruowanie odcinków AA' , BB' i CC' , które spotykają się w jednym punkcie. Jeśli punkt ten należy do obszaru trójkąta ΔABC , będzie punktem Fermata.

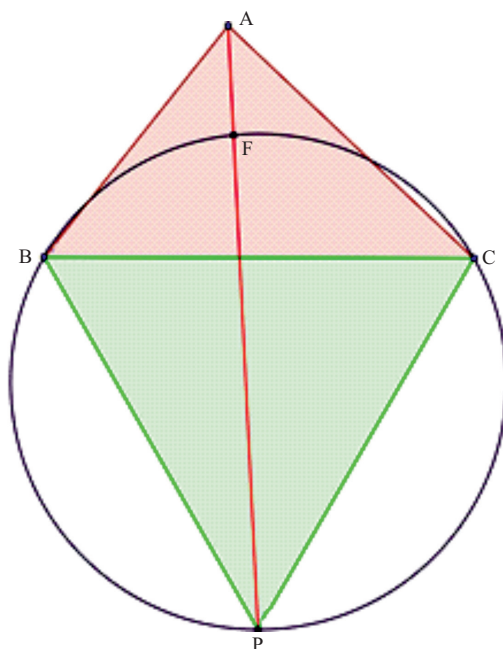


Rys. 4. Problem punktu Fermata z oryginalnym rozwiązaniem

Torricelli odpowiedział innym, równoważnym rozwiązaniem takim, aby (rys. 5):

- skonstruować punkt P taki, że $\triangle BCP$ jest trójkątem równobocznym i nie przecina się z $\triangle ABC$, P jest nazwany punktem zastępczym Torricellego dla pary punktów B i C,
- skonstruować okrąg opisany na trójkącie $\triangle BCP$, a krótki łuk BC tego okręgu nazwany jest łukiem Torricellego dla pary punktów BC,
- jeśli odcinek AP przecina się z łukiem Torricellego dla BC, to punkt przecięcia będzie punktem Fermata.

Rozwiązanie Torricellego jest bardziej użyteczne dla dalszego rozwiązania problemów EST(n) dla $n > 3$. Rozwiązanie to bazuje na fakcie, że dla każdego punktu T należącego do łuku Torricellego, następujące równanie jest prawdziwe: $|TP| = |TB| + |TC|$. W związku z tym suma $|TA| + |TB| + |TC|$ będzie najmniejsza, jeśli suma $|TA| + |TP|$ zostaje zminimalizowana, co zostanie spełnione, jeśli T należy do AP.

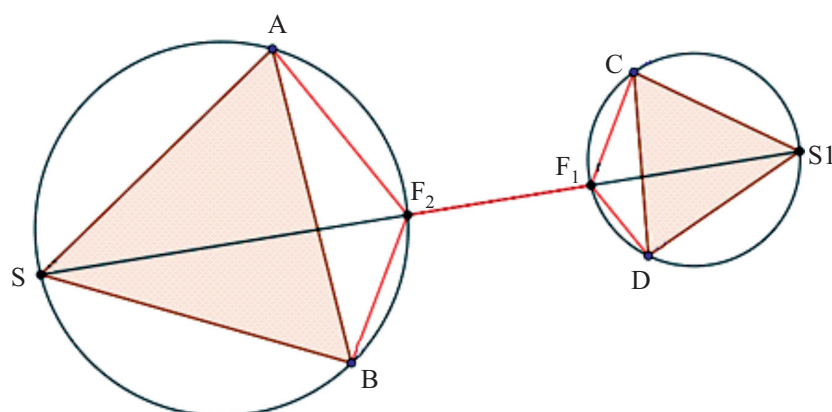


Rys. 5. Rozwiązanie Torricellego dla problemu punktu Fermata

Oba rozwiązania Fermata i Torricellego są poprawne tylko dla trójkątów z kątami mniejszymi niż 120° . Dla trójkątów z większymi kątami punktem F będzie wierzchołek trójkąta z największym kątem. Problem EST(4) może być rozwiązany następująco (rys. 6). Należy:

- wybrać dwa punkty spośród czterech podanych i oznaczyć wybrane punkty jako A i B, a pozostałe jako C i D,
- skonstruować punkt zastępczy Torricellego i łuk Torricellego dla pary A, B: $S = S(A,B)$ oraz $\alpha = \alpha(A,B)$,
- skonstruować punkt Fermata dla C, D i S: $F_1 = F(C,D,S)$,
- jeśli α przecina SF_1 w F_2 , to istnieje rozwiązanie problemu EST składające się z odcinków AF_2 , BF_2 , F_1F_2 , CF_1 and DF_1 ,
- w przeciwnym przypadku należy wrócić do pierwszego kroku i wybrać inną parę A, B.

Udowodniono, że poprawny wynik istnieje dla przynajmniej jednej pary punktów A, B.



Rys. 6. Rozwiązanie dla EST(4)

Zaproponowana metoda dla EST(4) może być uogólniona dla EST(n) z $n > 4$, następująco. Należy:

- wybrać dwa punkty wśród n podanych punktów i oznaczyć wybrane punkty A i B,
- skonstruować punkt zastępczy Torricellego i łuk Torricellego dla pary A, B: $S = S(A, B)$ oraz $\alpha = \alpha(A, B)$,
- rozwiązać problem EST($n-1$) dla zbioru punktów składających się z S i pozostałych punktów. Niech F_1 będzie pierwszym punktem, poprzez który S jest połączony z pozostałymi punktami w rozwiązaniu EST($n-1$),
- jeśli α przecina SF_1 w F_2 , to mamy rozwiązanie problemu EST(n) składające się z odcinków AF_2 , BF_2 oraz zbioru elementów rozwiązania EST($n-1$) po odjęciu odcinka SF_2 ,
- w przeciwnym przypadku należy wrócić do pierwszego kroku i wybrać inną parę A, B.

Podany algorytm ma złożoność $O(n!)$ i ma zastosowanie w praktyce dla mniejszych wartości n . Dla większych wartości n znalezienie EST za pomocą tego algorytmu w ograniczonym czasie jest niemożliwe. Pomimo tego ważna właściwość tego algorytmu będzie wykorzystana w proponowanym rozwiązaniu do skonstruowania zbioru punktów kandydujących. Otóż punkt kandydujący musi być:

- punktem zastępczym Torricellego dla pary punktów, wśród których każdy jest punktem końcowym lub innym punktem kandydującym, albo
- punktem Fermata dla trójki punktów, wśród których każdy jest punktem końcowym lub innym punktem kandydującym.

Inny udowodniony fakt jest taki, że liczba punktów Steinera nie może przekroczyć $n - 2$ [9].

W związku z powyższym można generować zbiór punktów kandydujących w sposób jak na rys 7.

```

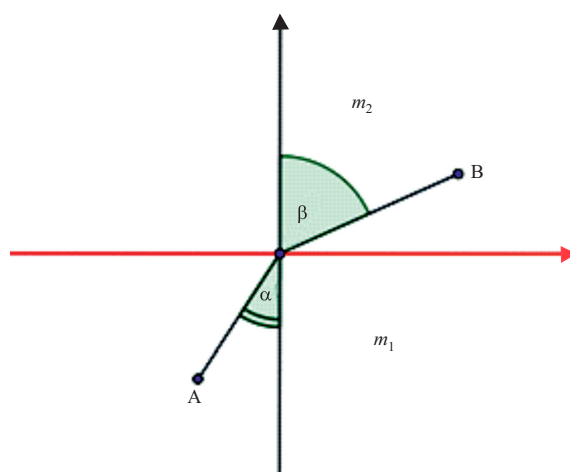
procedure WygenerujPunktyKandydująceEST
(in punktyKońcowe; out punktyKandydujące)
begin
  punktyKandydujące = punktyKońcowe
  n = Liczebność (punktyKońcowe)
  for i=1 to n-2 do
    begin
      punktyKandydujące += {zbiór punktów,
        z których każdy jest punktem
        zastępczym Torricellego dla pary
        punktów w zbiorze punktyKandydujące}
      punktyKandydujące += {zbiór punktów,
        z których każdy jest punktem
        Fermata dla trójki punktów w zbiorze
        punktyKandydujące}
    end
  punktyKandydujące -= punktyKońcowe
end

```

Rys. 7. Sposób generowania zbioru punktów kandydujących

Do tej pory szukano punktów kandydujących dla EST. Dla FNO potrzebny jest większy zbiór punktów kandydujących, z powodu wpływu funkcji g określonej kolekcją g -wielokątów. Każdy g -wielokąt ma przypisaną wartość kosztu. Istnienie g -wielokątów powoduje, że linia prosta pomiędzy dwoma punktami nie zawsze jest linią najmniejszego kosztu.

Rozważmy sytuację pokazaną na rys. 8, gdzie płaszczyzna jest podzielona na dwie półpłaszczyzny z różnymi kosztami (m_1 i m_2). Trzeba znaleźć linię najmniejszego kosztu, który połączy punkty A i B należące do dwóch półpłaszczyzn.



Rys. 8. Zasada Fermata

Według zasady Fermata, linią najmniejszego kosztu jest łamana A-O-B, gdzie O jest punktem granicy pomiędzy dwoma półpłaszczyznami i spełnia następujący warunek:

$$\frac{\sin \alpha}{\sin \beta} = \frac{m_1}{m_2} \quad (3)$$

Zasada Fermata ma szerokie zastosowanie w fizyce, m.in. wyjaśnia zjawisko refrakcji światła. Punkt O nazywamy punktem refrakcji dla pary punktów A i B przez linię graniczną.

W FNO zasada Fermata pozwala na znalezienie nowego podzbioru punktów kandydujących, każdy z nich jest punktem refrakcji dla pary innych punktów kandydujących przez krawędź g -wielokąta. Biorąc pod uwagę punkty refrakcji zaproponowano procedurę generowania zbioru punktów kandydujących dla FNO jak na rys. 9.

```

procedure WygenerujPunktyKandydująceFNO
  (in źródła, docelowe,  $g$ -wielokąty; out punktyKandydujące)
begin
  punktyKandydujące = źródła + docelowe +
    {zbiór wierzchołków  $g$ -wielokątów }
   $n$  = Liczebność (źródła + docelowe)
  for  $i=1$  to  $n-2$  do
    begin
      punktyKandydujące += {zbiór punktów,
        z których każdy jest punktem
        zastępczym Torricellego dla pary
        punktów w zbiorze punktyKandydujące}
      punktyKandydujące += {zbiór punktów,
        z których każdy jest punktem
        Fermata dla trójki punktów w zbiorze
        punktyKandydujące}
      punktyKandydujące += {zbiór punktów,
        z których każdy jest punktem
        refrakcji dla pary punktów należących
        do punktyKandydujące przez krawędź  $g$ -
        wielokąt}
    end
  punktyKandydujące -= źródła + docelowe;
end

```

Rys. 9. Procedura generowania zbioru punktów kandydujących dla FNO

Liczebność zbioru punktów kandydujących wzrasta zgodnie z $O((n+m)^{3n})$, gdzie n to liczba punktów źródłowych i docelowych, zaś m to sumaryczna liczba wierzchołków g -wielokątów.

4.2. Transformacja problemu FNO do postaci ILP

Niech K oznacza w problemie FNO zbiór punktów kandydujących wygenerowanych dla podanych zbiorów S , D i g -wielokątów, a V oznacza sumę S , D , T_1 i K . Dla każdej pary $u, v \in V$ definiowana jest zmienną binarną $x_{uv} \in \{0,1\}$ i obliczany koszt instalacji kabli wzdłuż odcinka $u-v$ za pomocą całki funkcji g :

$$c_{uv} = \int_{r=0}^l g((1-r)u + rv) dr \quad (4)$$

Ponieważ algorytm dokładny nie jest głównym celem tej pracy, warunki związane z przepustowością są pomijane.

Problem FNO może być transformowany do postaci ILP, oznaczonej $ILP_FNO(V, S, D, c)$, następująco:

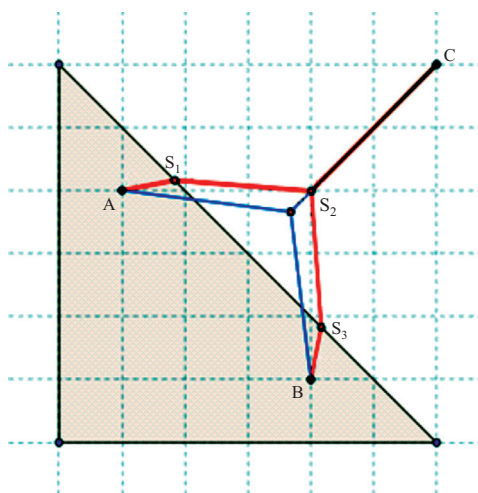
Zminimalizować: $\sum_{u,v \in V} c_{uv} x_{uv}$ z warunkami: $\sum_{u \in M, v \in V \setminus M} x_{uv} \geq 1$ dla każdego zbioru

$M \subset V, M \cap S \neq \emptyset$ i $V \setminus M \cap D \neq \emptyset$.

ILP_FNO może być rozwiązany za pomocą komercyjnych pakietów optymalizacyjnych. Dodatkowo ILP_FNO ma bardzo podobną formę do sformułowania ILP dla problemu TST [14]. W szczególności

można go przekształcić ILP_FNO do postaci ILP dla TST poprzez wprowadzenie „super źródła” łączącego wszystkie elementy zbioru S poprzez łącze z zerowym kosztem. Żeby znaleźć dokładne lub aproksymacyjne rozwiązanie problemu ILP_FNO, można więc zastosować algorytmy zaproponowane dla TST.

Dla instancji FNO (Przykład 1) przedstawionej na rys. 10, gdzie istnieje tylko jeden g -wielokąt w postaci brązowego trójkąta z kosztem równym 2, algorytm dokładny zwraca optymalny wynik w postaci czerwonych odcinków: AS₂, BS₃, CS₁, S₁S₂, S₁S₃, z sumarycznym kosztem równym 10,59. Gdy g -wielokąty byłyby nie uwzględnione, suma kosztu byłaby równa 10,96 (o 3,5% droższy).



Rys. 10. Przykład 1 – FNO do rozwiązania przez algorytm dokładny

Ponieważ problem FNO jest NP-trudny, utworzony algorytm dokładny pozwala na znalezienie rozwiązania tylko dla małych instancji FNO. Autor dokonał implementacji tego algorytmu i przeprowadził testy wydajnościowe, z których wynika, że dla typowych konfiguracji systemów komputerowych stosowanych do planowania sieci FTTH możliwe będzie zastosowanie algorytmu dokładnego dla sieci obejmujących nie więcej niż 10 punktów końcowych (reprezentujących użytkowników sieci). Takie rozwiązanie jest nieprzydatne w praktyce, gdyż projektowane fragmenty sieci obejmują zazwyczaj co najmniej kilkuset użytkowników.

5. Algorytm aproksymacyjny dla FNO

W sekcji 4 wykorzystano właściwości matematyczne EST w celu znalezienia punktów kandydujących do optymalizacji. Podejście to pozwala na znalezienie dokładnego optymalnego rozwiązania, ale ze względu na ogromną liczebność zbioru punktów kandydujących, zastosowanie tego algorytmu w przypadku większej sieci nie jest możliwe w rozsądnym czasie.

5.1. Strategia algorytmu aproksymacyjnego

W przeciwieństwie do algorytmu dokładnego, który rozpoczyna się już z pełnym zbiorem punktów kandydujących, algorytm aproksymacyjny zaczyna się z podstawowym zbiorem punktów kandydujących składającym się tylko z podanych punktów: źródeł, docelowych, istniejących punktów tranzyto-

wych i wierzchołków g -wielokątów. Po rozwiązaniu ILP_FNO dla tych punktów poprawimy wynik poprzez iteracje dodawania nowych punktów do zbioru punktów kandydujących. Liczba iteracji poprawienia wyników jest zależna od parametrów algorytmu. Głównymi parametrami konfiguracyjnymi algorytmu są:

- Rozdzielczość Luźnych Wierzchołków (*Loose Vertices Resolution, LVR*). Parametr ten określa liczbę powtórzeń, w których znajdujemy punkty Fermata dla każdej trójki punktów kandydujących leżących na tym samym g -wielokącie. W ten sposób można poprawić jakość wyniku lokalnie;
- Cięcia krawędzi wielokąta (*Polygon Edge Cut, PEC*). Liczba powtórzeń, na których znajdujemy punkty quasi-refrakcyjne i dodajemy je do zbioru punktów kandydujących.

Dla implementacji ILP_FNO, oprócz wykorzystania komercyjnych pakietów optymalizacyjnych, został utworzony także własny algorytm bazujący na znanych algorytmach dla problemu Topologicznego Drzewa Steinera [15]–[16]. Szczegóły algorytmu aproksymacyjnego są przedstawione na rys. 11.

```

procedure ObliczWektorKosztu
(in punktyKandydujące; out wektorKosztów)
begin
  foreach v1,v2 in punktyKandydujące do
    begin
      wektorKosztów[v1,v2] = koszt przypisany
      do  $g$ -wielokąta zawierającego v1 i v2.
    end
  end
end

procedure ObliczILP_FNO(in punktyKandydujące,
  źródła, docelowe, wektorKosztów; out
  steiner_points, steiner_lines)
begin
  Rozwiązać ILP za pomocą CPLEX, MS Solver
  lub algorytmu Steiner
end

procedure GłównyAlgorytmFNO (in PEC, LVR; in
  źródła, docelowe, istniejąceTranzytowe,
   $g$ -wielokąty; out steiner_points,
  steiner_lines)
begin
  punktyKandydujące = źródła + docelowe +
  istniejąceTranzytowe +
  wierzchołki( $g$ -wielokąty)
  for i=1 to PEC do
    begin
      foreach a, b in punktyKandydujące do
        begin
          if a, b należy do różnych  $g$ -wielokątów
          then
            begin
              Skonstruować punkty cięcia odcinka (A,
              B) z  $g$ -wielokątami. Nazywamy je x1,
              x2, .. xn
              punktyKandydujące += {x1, x2, .. xn}
            end
          end
        end
      for j=1 to LVR do
        begin

```

```

foreach a, b, c in punktyKandydujące do
begin
  if a, b, c należy do tego samego
    g-wielokąta then
  begin
    Skonstruować punkt Fermata dla a,
      b, c. Nazywamy go f
    punktyKandydujące += {f}
  end
end
end

wektorKosztów = ObliczWektorKosztu
(punktyKandydujące)
ObliczILP_FNO(punktyKandydujące, źródła,
docelowe, wektorKosztów,
steiner_points, steiner_lines)
foreach s in steiner_points do
begin
  if s należy do krawędzi g-wielokąta
  then
  begin
    Znajdź dwa punkty kandydujące
      najbliższe (w prawej i lewej
      stronie) do s w granicy
      g-wielokąta: v1 i v2
    Ustaw punkty środkowe odcinków: v1-s
      i v2-s, nazywamy je x1 i x2
    punktyKandydujące += {x1,x2}
  end
end
end
end

```

Rys. 11. Szczegóły algorytmu aproksymacyjnego

Złożoność algorytmu aproksymacyjnego wynosi $O(n^4) \cdot \text{PEC} \cdot \text{LVR} + S(n^{4 \cdot \text{PEC}}) \cdot \text{PEC}$, gdzie:

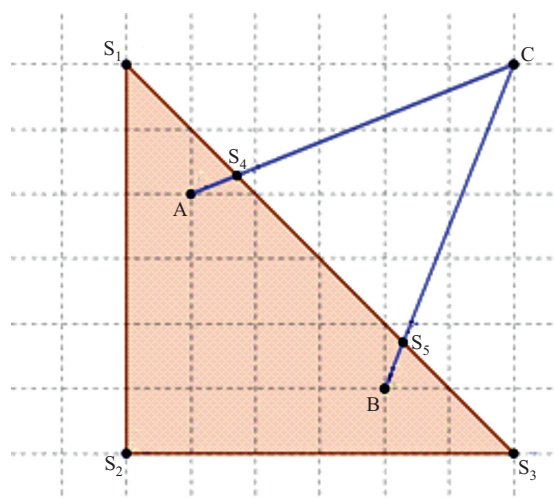
- n jest liczebnością początkowego zbioru punktów kandydujących (składającego się z S , D i wierzchołków g -wielokątów),
- $S(k)$ jest złożonością algorytmu rozwiązania problemu Topologia Drzewa Steinera dla wykresu k -elementów.

6. Ocena jakości wyników i wydajności algorytmu FNO i porównanie z algorytmem dokładnym

W celu porównania działania algorytmów aproksymacyjnego i dokładnego rozważamy ponownie instancję FNO w Przykładzie 1, który został rozwiązany wcześniej przez algorytm dokładny (rys. 10).

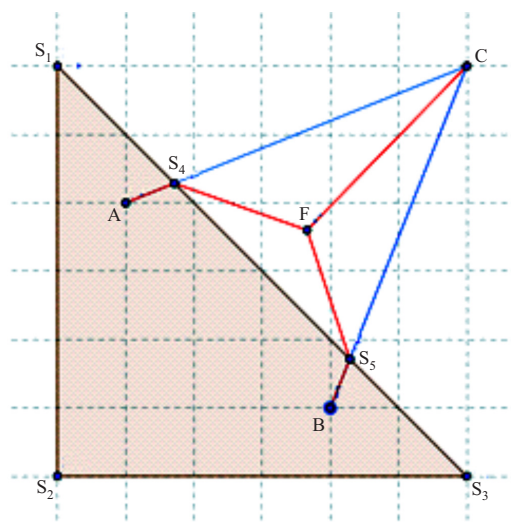
Algorytm działa w następujących krokach:

1. Rozpoczyna się od podstawowych punktów kandydujących A, B, C (dane punkty), S1, S2 i S3 (wierzchołki g -wielokąta).
2. Dodaje do zbioru punktów kandydujących punkty cięcia AC i AB z krawędziami g -wielokąta: S4 i S5 (rys. 12).



Rys. 12. Działanie algorytmu aproksymacyjnego dla Przykładu 1 (krok 2)

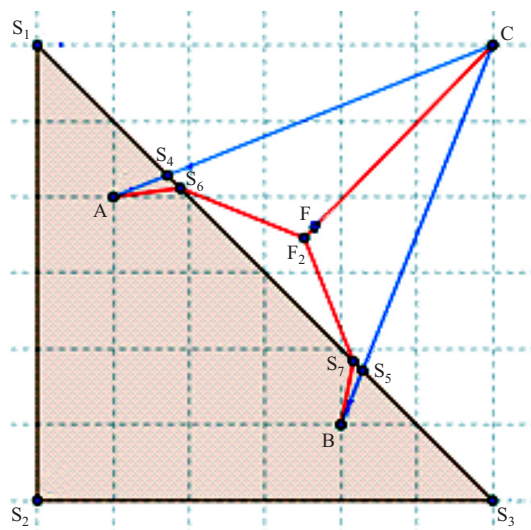
- Konstruuje punkty Fermata dla wszystkich trójek punktów należących do tego samego g -wielokąta. W tym przykładzie najbardziej interesujący punkt Fermata to ten dla trójki (C, S4 i S5).
- Rozwiązuje ILT_FNO dla (A, B, C, K, S1, ..., S5). Wynikiem jest czerwona łamana przedstawiona na rys. 13.



Rys. 13. Działanie algorytmu aproksymacyjnego dla Przykładu 1 (krok 4)

- Konstruujemy punkty środkowe odcinków utworzonych przez punkty kandydujące należące do krawędzi g -wielokąta (S_1S_4 , S_4S_5 , itp.). Są traktowane je jako quasi-refrakcyjne punkty. Dodaje je się do zbioru punktów kandydujących.
- Zmniejsza się PEC o jeden, a następnie powtarza krok 2 aż do momentu, gdy PEC wynosi zero.

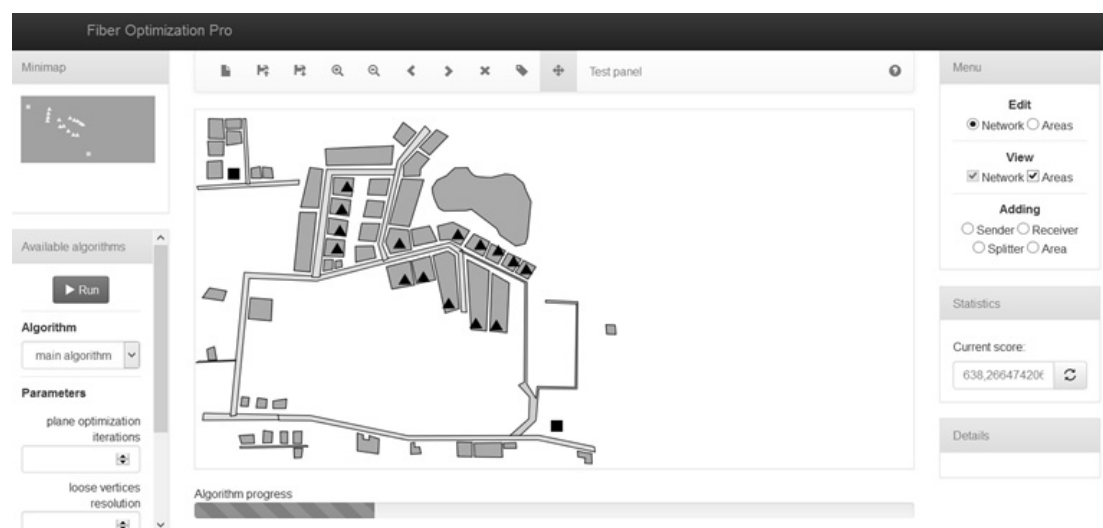
Wynik końcowy osiągnięty dla $PEC = 20$ przedstawiono na rys. 14. Wynik ten jest bardzo bliski do wyniku algorytmu dokładnego (0,001% różnicy kosztów).



Rys. 14. Działanie algorytmu aproksymacyjnego dla Przykładu 1 – wynik końcowy

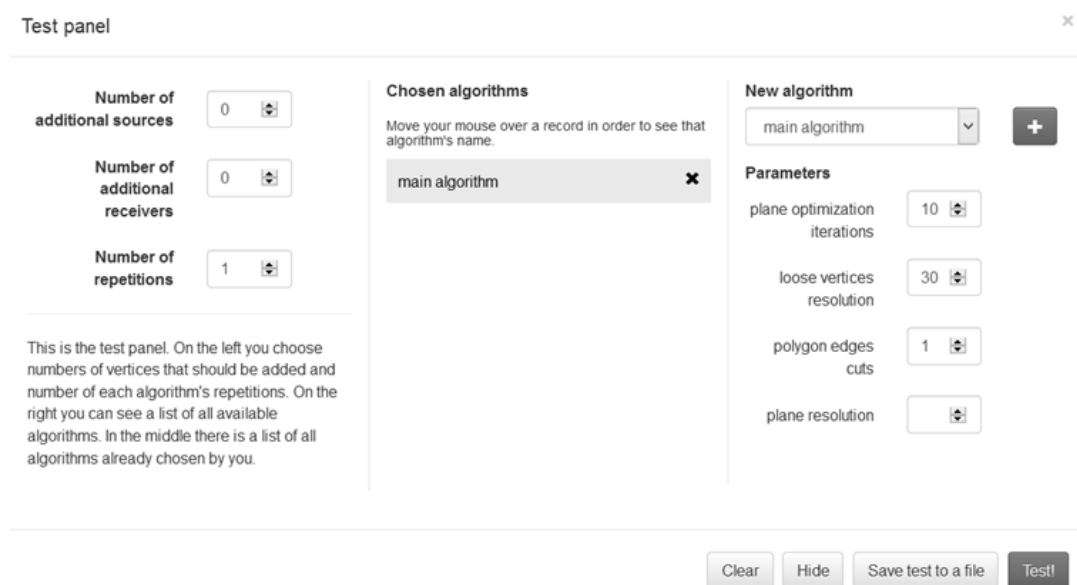
7.1. Analiza skuteczności działania algorytmu na przykładzie

W celu oceny skuteczności działania algorytmu zostało on zaimplementowany w języku .net C#. Program został uruchomiony na komputerze PC z procesorem Intel Core i5 2,66 GHz i 8 GB RAM.



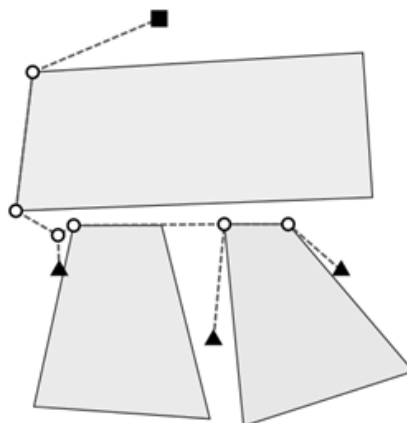
Rys. 15. Implementacja algorytmu FNO

Aplikacja ta zawiera panel testowania, który pozwala na masowe testowanie algorytmu dla różnych wartości parametrów PEC and LVR (rys. 16).



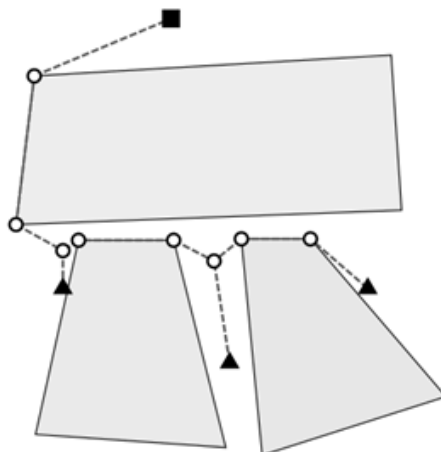
Rys. 16. Implementacja algorytmu FNO – panel testowy

W celu analizy, w jaki sposób parametry PEC i LVR mają wpływ na jakość wyniku i zużycie czasu, rozważmy instancję FNO w Przykładzie 2, w którym mamy jedno źródło (symbol kwadratu), trzy przeznaczenia (trójkąty), do których ma źródło być podłączone, oraz trzy wielokąty z wysokim kosztem.



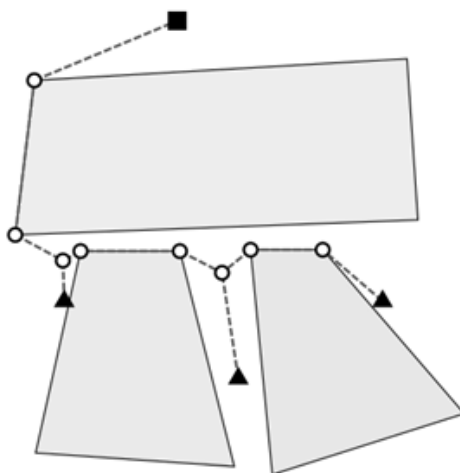
Rys. 17. Wynik algorytmu dla Przykładu 2, $PEC = 1$, $LVR = 1$, koszt = 644,9

Dla $PEC = 1$ i $LVR = 1$ (rys. 17) koszt jest dość wysoki (644,9), ale algorytm działa wyjątkowo szybko – 0,7 s.



Rys. 18. Wynik algorytmu dla Przykładu 2, wynik dla $PEC = 1$, $LVR = 10$, koszt = 638.3

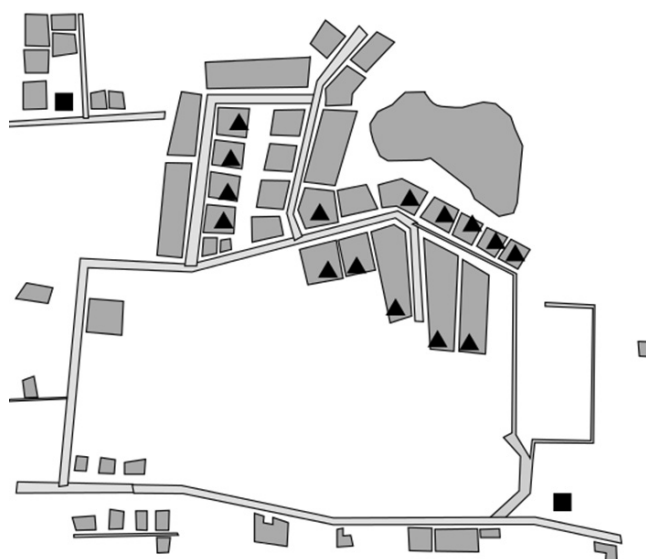
Po zwiększeniu LVR do 10 (rys. 18), koszt znacząco się zmniejszy. Algorytm działa wolniej (6 s), ale jego szybkość nadal jest akceptowalna (czas prawie niezauważalny).



Rys. 19. Wynik algorytmu dla Przykładu 2, wynik dla $PEC = 1$, $LVR = 20$, koszt = 638.2

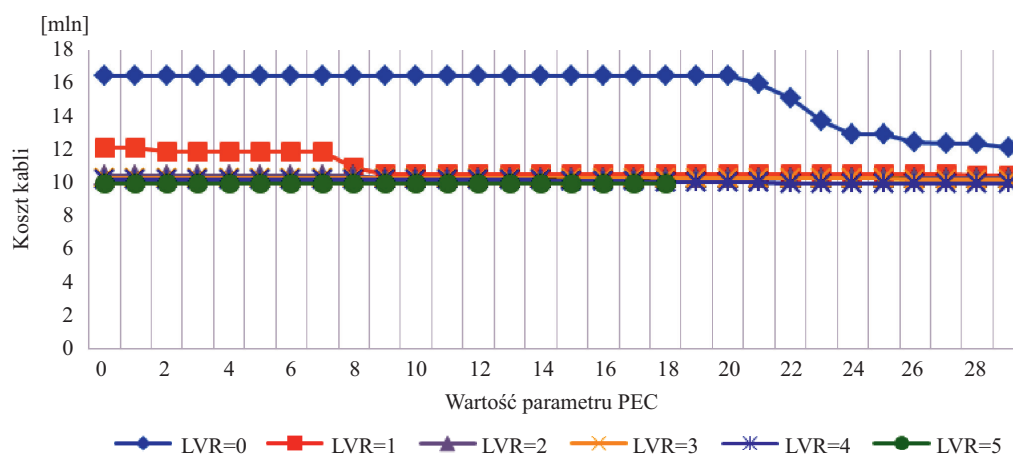
Jeśli zwiększymy LVR do 20 (rys. 19), koszt minimalnie się obniży (do 638,2). Algorytm działa jednak wówczas dużo wolniej (374 s) i jest to szybkość nie do przyjęcia w praktycznym narzędziu do projektowania sieci.

W szczegółowej analizie badano wpływ parametrów konfiguracyjnych PEC i LVR na wyniki algorytmu za pomocą testu masowego w praktycznym przykładzie optymalizacji sieci FTTH dla jednego z osiedli w Warszawie (Przykład 3, rys. 20). Test masowy polega na wielokrotnym uruchomieniu algorytmu dla różnych wartości parametrów konfiguracyjnych dla poszczególnych iteracji testu.



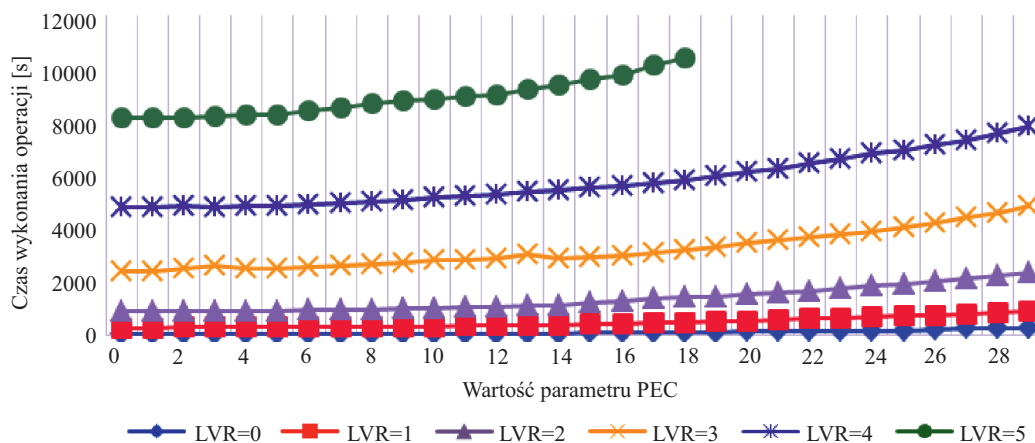
Rys. 20. Przykład 3 – badanie dla osiedla w Warszawie

Na rys. 21 przedstawiony jest wykres tendencji zmiany jakości wyników wraz ze wzrostem parametrów PEC i LVR. Zaobserwowano, że zwiększenie LVR z 0 na 1 powoduje znaczącą poprawę jakości wyników. Zwiększenia LVR z 1 na 2 przynosi już tylko umiarkowaną poprawę. Natomiast dalsze zwiększenie tego parametru już nie przynosi zauważalnego efektu. Tendencja ta jest logiczna, gdyż parametr LVR decyduje jedynie o stopniu lokalnego poprawienia wyników algorytmu polegającym na znalezieniu punktów Fermata. Natomiast wpływ PEC na działanie algorytmu jest inny. Zaobserwowano, że wymagana jest pewna liczba iteracji ($PEC = 20$ dla $LVR = 0$ i $PEC = 7$ dla $LVR = 1$), żeby następowała znacząca poprawa wyników. Tendencja ta jest logiczna, gdyż parametr PEC decyduje o liczebności zbioru punktów kandydujących, a algorytm zaczyna zwracać wyniki zbliżone do wyniku optymalnego dopiero w momencie, gdy zbiór punktów kandydujących posiada wystarczająco dużą liczbę elementów.



Rys. 21. Przykład 3 – jakość wyników w zależności od parametrów PEC i LVR

Wydajność algorytmu jest prezentowana na rys. 22. Tendencja zmian wykazuje, że każde zwiększenie parametru LVR wiąże się ze znaczącym wzrostem czasu wykonania operacji, zaś wzrost w parametrze PEC powoduje raczej łagodny wzrost tego czasu.



Rys. 22. Przykład 3 – zależność wydajności od parametrów PEC i LVR

Jako wniosek z obserwacji testów masowych zaleca się w sytuacjach praktycznych uruchomienie algorytmu z wartościami LVR w okolicy 1, 2 lub 3, zaś parametr PEC należy zwiększyć maksymalnie, jak to możliwe.

Przeprowadzone testy masowe potwierdzają, że zaproponowany algorytm aproksymacyjny, za cenę rezygnacji z poszukiwania rozwiązania idealnego dla problemu FNO, pozwoli na wyznaczenie rozwiązania bliskiego optimum, ograniczając jednocześnie złożoność obliczeniową i umożliwiając zastosowanie algorytmu w systemach komputerowych o wielkości i cenie akceptowalnej dla przedsiębiorstw zajmujących się projektowaniem, budową i eksploatacją sieci światłowodowych.

7. Podsumowanie

W artykule przedstawiona została nowa metoda optymalizacji sieci FTTH koncentrująca się na minimalizacji kosztu instalacji kabli. Metoda ta jest rozwinięciem wyników dotychczasowych prac na temat optymalizacji geometrycznej. Dodane zostały nowe aspekty nieuwzględnione w dotychczasowych pracach, w szczególności aspekt uwarunkowań terenowych i aspekt istniejących zasobów sieci.

Zagadnienie optymalizacji sieci FTTH zostało sformułowane w postaci problemu matematycznego, któremu nadano roboczą nazwę *FTTH Network Optimization, FNO*. Problem FNO został następnie przekształcony do postaci problemu *Integer Linear Programming, ILP*.

Ponieważ problem FNO jest NP-trudny, utworzony algorytm dokładny ILP pozwala na znalezienie rozwiązania tylko dla małych instancji FNO. Algorytm został zaimplementowany i przeprowadzone zostały testy wydajnościowe, z których wynika, że dla typowych konfiguracji systemów komputerowych stosowanych do planowania sieci FTTH możliwe będzie zastosowanie algorytmu dokładnego dla sieci obejmujących nie więcej niż 10 punktów końcowych (reprezentujących użytkowników sieci). Takie ograniczenie jest nieakceptowalne w praktyce, gdyż projektowane fragmenty sieci obejmują zazwyczaj co najmniej kilkuset użytkowników.

W związku z powyższym, rozwiązaniem problemu dla większych instancji FNO będzie algorytm aproksymacyjny, który za cenę rezygnacji z poszukiwania rozwiązania idealnego, pozwoli na wyznaczenie rozwiązania bliskiego optimum, ograniczając jednocześnie złożoność obliczeniową i umożliwiając zastosowanie algorytmu w systemach komputerowych o wielkości i cenie akceptowalnej dla przedsiębiorstw zajmujących się projektowaniem, budową i eksploatacją sieci światłowodowych.

W artykule został opisany algorytm aproksymacyjny rozwiązujący problem FNO. W celu potwierdzenia jego praktycznej skuteczności wykonano implementację w postaci programu komputerowego napisanego w języku .net C#. W celu weryfikacji jakości wyników algorytmu oraz oceny wpływu zmienności poszczególnych parametrów konfiguracyjnych przeprowadzono test masowy dla praktycznego przykładu sieci (zmieniając ustawienia parametrów konfiguracyjnych dla poszczególnych iteracji testu). Przedstawione zostało zalecenie odnośnie możliwości zastosowania algorytmu w praktyce.

Bibliografia

- [1] Williams B.: *FTTH Council Global Alliance Updates*, 2013 LATAM Conference, FTTH Council, 2013
- [2] Mazzali C.: *Optimization of FTTH passive optical networks continues*, Lightwave Magazine, January 2005
- [3] Chardy M., Hervet C.: *FTTH network design under OA&M constraints*, INOC'11 Proceedings of the 5th International Conference on Network Optimization, 2011
- [4] Loggerenberg S., Grobler L., Terblanche F.: *Optimization of PON Planning for FTTH Deployment Based on Coverage*, Proceedings of Southern African Telecommunications and Networks Access Conference (SATNAC), 2012
- [5] Grötschel M., Raack C., Werner A.: *Towards optimizing the deployment of optical access networks*, EURO Journal on Computational Optimization, vol. 2, no. 1–2, 2013
- [6] Chardy M., Costa M.-C., Faye A., Trampont M.: *Optimizing splitter and fiber location in a multi-level optical FTTH network*, European Journal of Operational Research, vol. 222, issue 3, 2012
- [7] Hervet C., Faye A., Costa M., Chardy M., Francfort S.: *Solving the Two-Stage Robust FTTH network design Problem under Demand Uncertainty*, Electronic Notes in Discrete Mathematics, vol. 41, 2013
- [8] Kadhim D., Hussain N.: *Link and Cost Optimization of FTTH Network Implementation through GPON Technology*, Communications and Network, vol. 5, 2013
- [9] Brazil M.: *On the History of the Euclidean Steiner Tree Problem*, Archive for History of Exact Sciences, vol. 68, no. 3, 2014
- [10] Arora S.: *Polynomial time approximation schemes for Euclidean traveling salesman and other geometric problems*, Journal of the ACM, vol. 45, no. 5, 1998
- [11] Chung-Wei L.: *Efficient Obstacle-Avoiding Rectilinear Steiner Tree Construction*, ISPD'07, March 18–21, 2007
- [12] Zachariasen M., Winter P.: *Obstacle-Avoiding Euclidean Steiner Trees in the Plane: An Exact Algorithm*, *Algorithm Engineering and Experimentation*, Lecture Notes in Computer Science, vol. 1619, 1999.

- [13] Garey M.: *The Complexity of Computing Steiner Minimal Trees*, SIAM Journal of Applied Mathematics, vol. 32, no. 4, June 1977.
- [14] Laarhoven J. W.: *Exact and heuristic algorithms for the Euclidean Steiner tree problem*, PhD thesis, University of Iowa, 2010
- [15] Mehlhorn K.: *A faster approximation algorithm for the Steiner problem in graphs*, Information Processing Letters, vol. 27, no. 3, 1998
- [16] Vazirani V.: *Approximation Algorithms*, Springer-Verlag, Berlin, 2001

Hoang Nghia Le

Hoang Nghia Le ukończył studia magisterskie na wydziale Elektroniki i Telekomunikacji Politechniki Warszawskiej w 1998 roku. Obecnie jest menadżerem produktu w firmie Suntech S.A, spółki specjalizującej się w produkcji systemów OSS dla sieci telekomunikacyjnych. Jego naukowe zainteresowania obejmują zagadnienia optymalizacji sieci optycznych i bezprzewodowych

e-mail: NghiaLe.Hoang@suntech.pl