

mgr Andrzej Pisarski,
Wydział Fizyki,
Uniwersytet w Białymstoku

ALGORYTM ZNAJDOWANIA PROMIENIA POKRYCIA SIECI FILTRÓW W PROBLEMIE DETEKcji SŁABYCH PRAWIE MONOCHROMATYCZNYCH SYGNAŁÓW

Jednym z problemów pojawiających się podczas konstrukcji efektywnych pokryć sieciowych przestrzeni parametrów sygnałów w detekcji słabych i prawie monochromatycznych sygnałów w szumie jest znalezienie promienia pokrycia sieci, czyli minimalnego promienia sfer, których środki są zaczepione w węzłach sieci i które dają pokrycie przestrzeni bez dziur. W pracy przedstawiony został algorytm numerycznego znajdowania promienia pokrycia dla sieci o węzłach wyznaczanych przez zadane wektory bazowe.

One of the problems in the design of effective banks of templates in the detection of weak and almost monochromatic signals in noise is to find covering radius of a given grid, i.e. the minimum radius of the spheres which centered at the nodes of the grid cover all space without holes. We present a numerical algorithm for finding the covering radius of the grid defined by its base vectors.

1. Wstęp

Problem detekcji słabych prawie monochromatycznych sygnałów pojawia się m.in. w analizie danych zbieranych przez obecnie pracujące interferometryczne detektory fal grawitacyjnych VIRGO¹, LIGO² czy GEO600³. Spodziewamy się, że gdy detektory osiągną zakładaną czułość, w zbieranych przez nie danych będzie można wykryć sygnał wywołany przejściem przez detektor fali grawitacyjnej pochodzącej m.in. od rotujących gwiazd neutronowych w naszej Galaktyce. Ponieważ amplituda rejestrowanego przez detektor sygnału wywołanego przejściem przez detektor fali grawitacyjnej jest wielokrotnie mniejsza od amplitudy szumu występującego w detektorze, to do jego wykrycia stosujemy teorię statystycznego wykrywania sygnałów w szumie.

W stosowanej przez nas *metodzie filtru dopasowanego*, wykrycie sygnału polega na (mówiąc w pewnym uproszczeniu) filtrowaniu danych tym samym sygnałem, który spodziewamy się wykryć. W przypadku wykrywania prawie monochromatycznych sygnałów metoda ta sprowadza się do problemu ekstremalizacji funkcji zwanej F-statystyką ze względu na tzw. parametry fazowe sygnału: bezwymiarowej częstotliwości fali, pierwszej pochodnej częstości oraz po dwóch parametrach i związanych z położeniem źródła fali grawitacyjnej na sferze niebieskiej (tzw. all-skysearches). Bezwymiarowe parametry i definiujemy następująco:

$$\omega_0 := 2\pi f_0 T_0, \quad 1.1. a$$

$$\omega_1 := \pi \dot{f}_0 T_0^2 \quad 1.1. b$$

gdzie jest częstotliwością sygnału wziętą w chwili $t=0$. Ponieważ ekstremalizację tę możemy przeprowadzić tylko w sposób numeryczny F-statystykę wyliczamy dla wielu

¹ <http://www.ego-gw.it/virgodescription/>

² <http://www.ligo.caltech.edu/>

³ <http://www.geo600.org/>

możliwych wartości parametrów sygnału. Wartości parametrów, dla których obliczamy F-statystykę tworzą dyskretny zbiór punktów - sieć -w przestrzeni parametrów sygnału (zwanej też przestrzenią filtrów; jest to przestrzeń kartezjańska, o osiach związanych z parametrami fazowymi sygnału). Wartości F-statystyki porównujemy z tzw. progiem. Jeśli dla pewnych wartości parametrów F-statystyka przekroczy próg to wartości te są estymatorami prawdziwego (wytworzonego przez przyrodę) sygnału.

Jednak przestrzeni parametrów sygnału nie można próbkować dowolnie gęsto gdyż wiąże się to z dużym kosztem czasowym związanym z obliczeniem wszystkich F-statystyk. W celu wyznaczenia optymalnego sposobu próbkowania przestrzeni parametrów posługujemy się poziomiami przybliżonej funkcji autokowariancji F-statystyki otrzymanej z rozwinięcia w szereg Taylora dokładnej funkcji autokowariancji F-statystyki (pełna autokowariancja F-statystyki jest niewygodna w użyciu ze względu na swą złożoność). Poziomice przybliżonej funkcji autokowariancji mają kształt powierzchni hiperelipsoidy której środkiem jest punkt należący do przestrzeni parametrów sygnału. Jeśli sygnał wyprodukowany przez przyrodę znajdzie się wewnątrz hiperelipsoidy to taki sygnał (z określonym prawdopodobieństwem) jesteśmy w stanie wykryć. Ponieważ zwykle istnieje potrzeba próbkowania przestrzeni parametrów sygnału o objętości znacznie większej niż objętość pojedynczej hiperelipsoidy, musimy pokryć przestrzeń parametrów sygnału wieloma przystającymi (nachodzącymi na siebie) hiperelipsoidami w taki sposób, aby dostać pokrycie przestrzeni bez dziur. Ograniczając się do pokryć sieciowych (tzn. takich w których punktami przestrzeni parametrów są węzły sieci otrzymane z kombinacji liniowej wektorów bazowych) efektywność pokrycia hiperelipsoidami przestrzeni parametrów można wyrazić za pomocą tzw. gęstości pokrycia:

$$\rho = \frac{\text{objętość hiperelipsoidy}}{\text{objętość hiperkomórki elementarnej}} \quad 1.2.$$

gdzie hiperkomórką elementarną jest 4-wymiarowy równoległoscian rozpięty na wektorach bazowych sieci. Im mniejsza gęstość pokrycia, tym bardziej efektywną sieć otrzymujemy (mniejszy koszt obliczenia F-statystyk dla wszystkich węzłów sieci). F-statystykę można obliczyć dla wielu częstotliwości na raz (ustalając parametry wszystkie poza częstotliwością) wykorzystując algorytm szybkiej transformaty Fouriera. Algorytm szybkiej transformaty Fouriera wymaga, aby węzły sieci w przestrzeni parametrów sygnału układały się wzdłuż prostych w taki sposób, aby odległość między dwoma sąsiednimi węzłami była równa odległości pomiędzy dwoma kolejnymi częstotliwościami Fouriera. W celu uproszczenia konstrukcji sieci filtrów problem pokrycia hiperelipsoidami zamienia się na problem pokrycia hipersferami. Zamiana ta jest możliwa dzięki wykorzystaniu odwzorowania, które przeprowadza wektor zaczepiony w początku układu współrzędnych oraz końcu na powierzchni hiperelipsoidy na wektor którego koniec znajduje się na powierzchni hipersfery o promieniu jeden. Odwzorowanie to działa na wszystkie wektory w tym i na wektor wiąz wynikający ze stosowania algorytmu szybkiej transformaty Fouriera. W wyniku otrzymujemy wektor wiąz o tym samym kierunku co przed przekształceniem (równoległy do osi częstości) lecz o nowej długości. Gęstość pokrycia w przestrzeni z hipersferami można obliczyć ze wzoru:

$$\rho(R, d) = \frac{2\pi^{d/2} R^d}{d \Gamma(d/2) |\det E|} \quad 1.3.$$

gdzie R – jest promieniem sfery w d -wymiarowej przestrzeni, E – macierzą wektorów bazowych (rozpinających komórkę elementarną), Γ – funkcją gamma Eulera.

W cyklu prac [1-5] pokazano, że dla czasów obserwacji nie dłuższych niż kilkanaście dni konstruując sieć filtrów można użyć uproszczonego modelu sygnału fali grawita-

cyjnej. Uproszczony model sygnału ma postać funkcji sinusoidalnej o stałej amplitudzie oraz fazie liniowej w parametrach fazowych:

$$h(t; h_0, \psi_0, \xi) = h_0 \sin(\Phi(t; \xi) + \psi_0), \quad 1.4.$$

gdzie h_0 – jest amplitudą sygnału, ψ_0 – fazą początkową, $\Phi(t; \xi)$ – fazą sygnału równą:

$$\Phi(t; \xi) = \omega_0 \frac{t}{T_0} + \omega_1 \left(\frac{t}{T_0} \right)^2 + \alpha_1 \mu_1(t) + \alpha_2 \mu_2(t), \quad 1.5.$$

gdzie $\mu_1(t)$ i $\mu_2(t)$ są znanymi funkcjami czasu opisującymi położenie detektora względem środka ciężkości Układu Słonecznego; jest czasem trwania obserwacji. Konstruując sieć filtrów wykorzystamy algorytm znajdowania promienia pokrycia, który zostanie omówiony bardziej szczegółowo.

2. Algorytm znajdowania promienia pokrycia sieci filtrów⁴

Niech konstruowana sieć filtrów nosi nazwę S2. Konstruując sieć S2 rozważymy deformację optymalnej, znanej w literaturze sieci. Środki hipersfer jednostkowych wyznaczają węzły sieci o następujących wektorach bazowych:

$$\mathbf{o1} = \left\{ \frac{1}{2\sqrt{2}}, -\frac{1}{2}\sqrt{5/6}, -\frac{1}{2}\sqrt{5/3}, -\frac{1}{2}\sqrt{5} \right\}; \quad 1.6. a$$

$$\mathbf{o2} = \left\{ \frac{1}{2\sqrt{2}}, \frac{1}{2}\sqrt{15/2}, 0, 0 \right\}; \quad 1.6. b$$

$$\mathbf{o3} = \left\{ \frac{1}{2\sqrt{2}}, -\frac{1}{2}\sqrt{5/6}, \sqrt{5/3}, 0 \right\}; \quad 1.6. c$$

$$\mathbf{o4} = \left\{ \frac{1}{2\sqrt{2}}, -\frac{1}{2}\sqrt{5/6}, -\frac{1}{2}\sqrt{5/3}, \frac{1}{2}\sqrt{5} \right\}; \quad 1.6. d$$

Wektory (1.6.a – 1.6.d) zaczepione w początku układu współrzędnych wyznaczają krawędzie boczne 4-wymiarowego sympleksu o podstawie będącej trójwymiarowym czworokątnym foremnym. W dalszej części pracy wektory te nazywać będziemy wektorami krawędziowymi. Długość wektora obliczamy w poniższy sposób:

$$\|[\mathbf{v}_a]_i\| := \text{Module}[\{\}, \text{Simplify}[\text{Sqrt}[\mathbf{v}_a \cdot \mathbf{v}_a]]] \quad 1.7.$$

Obliczyliśmy, że długość wszystkich wektorów krawędziowych wynosi $\sqrt{2}$. Lista wektorów krawędziowych:

$$\mathbf{lo} = \{\mathbf{o1}, \mathbf{o2}, \mathbf{o3}, \mathbf{o4}\}; \quad 1.8.$$

Oś ω_0 jest osią symetrii wektorów krawędziowych, tzn. Istnieje taki obrót (wokół osi ω_0), który przeprowadza jeden wektor bazowy w drugi. Z symetrii tej wynika, że składowe równoległe do osi ω_0 wszystkich wektorów krawędziowych są takie same. Rozważmy taką deformację sieci, która uzależnia wektory krawędziowe od dwóch parametrów: długości wektorów krawędziowych d oraz kąta α jaki tworzy dowolnie wybrany wektor krawędziowy z osią. Składowe wektorów bazowych równoległe do osi ω_0 :

$$\mathbf{hv} = \text{Apply}[\text{Plus}, \mathbf{lo}]/4; \quad 1.9.$$

Wektory krawędziowe rozpinają 4-wymiarowy sympleks o wysokości

$$h = \text{Sqrt}[\mathbf{hv} \cdot \mathbf{hv}] \quad 1.10.$$

tym samym wektor jednostkowy w kierunku osi ω_0 jest następujący:

⁴ Algorytm znajdowania pokrycia został zapiany w języku programu Mathematica. Wektory wyróżnione zostały czcionką pogrubioną.

$$\mathbf{hp} = \mathbf{hv}/h \quad 1.11.$$

Składowe wektorów krawędziowych w kierunku prostopadłym do osi ω_0 :

$$\mathbf{a1} = \mathbf{o1} - \mathbf{hv}; \quad 1.12. a$$

$$\mathbf{a2} = \mathbf{o2} - \mathbf{hv}; \quad 1.12. b$$

$$\mathbf{a3} = \mathbf{o3} - \mathbf{hv}; \quad 1.12. c$$

$$\mathbf{a4} = \mathbf{o4} - \mathbf{hv}; \quad 1.12. d$$

Wektory jednostkowe składowych prostopadłych do osi ω_0 :

$$\mathbf{a1p} = \mathbf{a1}/\|\mathbf{a1}\|; \quad 1.13. a$$

$$\mathbf{a2p} = \mathbf{a2}/\|\mathbf{a2}\|; \quad 1.13. b$$

$$\mathbf{a3p} = \mathbf{a3}/\|\mathbf{a3}\|; \quad 1.13. c$$

$$\mathbf{a4p} = \mathbf{a4}/\|\mathbf{a4}\|; \quad 1.13. d$$

Wymóg korzystania z algorytmu szybkiej transformaty Fouriera wymusza deformację sieci (1.6.a – 1.6.d) w taki sposób, aby otrzymać wektory bazowe z których jeden jest równoległy do osi ω_0 i ma długość $\Delta\omega'_0$. Pierwszy wektor bazowy (równoległy do osi ω_0) otrzymujemy sumując wszystkie cztery wektory krawędziowe. Na pozostałe wektory bazowe wybieramy ostatnie trzy wektory krawędziowe (1.6.b – 1.6.d). Wykorzystując więź $\Delta\omega'_0$ długość każdego z wektorów krawędziowych uzależniamy od kąta α :

$$dk[\alpha] := \quad 1.14.$$

gdzie $delp0prim = \Delta\omega'_0$.

Składowa równoległa do osi ω_0 , zależna od kąta α :

$$\mathbf{h2}[\alpha] := \quad 1.15.$$

Składowe prostopadłe do osi uzależnione od kąta α :

$$\mathbf{vb1}[\alpha] := \mathbf{h2}[\alpha] \quad 1.16 a$$

$$\mathbf{vb2}[\alpha] := \quad 1.16. b$$

$$\mathbf{vb3}[\alpha] := \mathbf{h} \quad 1.16. c$$

$$\mathbf{vb4}[\alpha] := \mathbf{h} \quad 1.16. c$$

W celu znalezienia promienia pokrycia sieci wyznaczamy wierzchołki 4-wymiarowej komórki elementarnej znajdując wszystkie możliwe kombinacje liniowe (o współczynnikach zero lub jeden) wektorów krawędziowych(1.16.a -1.16.d):

$$01 \quad \mathbf{k0} = \{0, 0, 0, 0\};$$

$$02 \quad \mathbf{k1}[\alpha] :=$$

$$03 \quad \mathbf{k2}[\alpha] :=$$

$$04 \quad \mathbf{k3}[\alpha] :=$$

$$05 \quad \mathbf{k4}[\alpha] :=$$

$$06 \quad \mathbf{k5}[\alpha] :=$$

$$07 \quad \mathbf{k6}[\alpha] :=$$

$$08 \quad \mathbf{k7}[\alpha] :=$$

$$09 \quad \mathbf{k8}[\alpha] :=$$

$$10 \quad \mathbf{k9}[\alpha] :=$$

$$11 \quad \mathbf{k10}[\alpha] :=$$

$$12 \quad \mathbf{k11}[\alpha] :=$$

$$13 \quad \mathbf{k12}[\alpha] :=$$

$$14 \quad \mathbf{k13}[\alpha] :=$$

$$15 \quad \mathbf{k14}[\alpha] := \mathbf{k}$$

$$16 \quad \mathbf{k15}[\alpha] :=$$

1.17.

Lista wszystkich szesnastu wierzchołków 4-wymiarowej komórki elementarnej:

$$\mathbf{kk}[\alpha] := \{\mathbf{k0}, \mathbf{k1}[\alpha], \mathbf{k2}[\alpha], \mathbf{k3}[\alpha], \mathbf{k4}[\alpha], \mathbf{k5}[\alpha], \mathbf{k6}[\alpha], \mathbf{k7}[\alpha], \mathbf{k8}[\alpha], \mathbf{k9}[\alpha], \mathbf{k10}[\alpha], \mathbf{k11}[\alpha], \mathbf{k12}[\alpha], \mathbf{k13}[\alpha], \mathbf{k14}[\alpha], \mathbf{k15}[\alpha]\}; \quad 1.18.$$

Poniżej utworzymy funkcję której celem będzie wyznaczenie najkrótszej odległości między wybranym punktem, wskazywanym przez wektor \mathbf{n} , a wierzchołkami komórki elementarnej danej przez wektory (1.17).

```

01  ss[a_, x_, y_, z_, u_, vp_] :=
02  Module[{qq, tt, v1, v2, v3, v4, kj},
03  v1 = k1[a];
04  v2 = k2[a];
05  v3 = k3[a];
06  v4 = k4[a];
07  kj = kk[a];
08  qq = Table[tt :=
09  Sqrt[tt.tt]], {i, 1, 16}];
10  {a, Min[qq], vp + (v1 x + v2 y + v3 z + v4u)}];

```

1.19

Funkcja przyjmuje sześć parametrów: kąt α oznaczony tutaj przez zmienną „a”, cztery parametry „x”, „y”, „z”, „u” odpowiadające za wybór punktu wskazywanego przez wektor \mathbf{n} oraz wektor \mathbf{vp} otrzymany z poprzednich iteracji programu opisanego poniżej. Wektor \mathbf{n} jest obliczany następująco:

$$\mathbf{n} = \mathbf{vp} + (\mathbf{v1} x + \mathbf{v2} y + \mathbf{v3} z + \mathbf{v4}u) \quad 1.20.$$

Wektor ten zapiszemy jeszcze raz w następującej postaci:

$$\mathbf{n}_m(\alpha) = \mathbf{n}_{m-1}(\alpha) + \frac{1}{3^{m-1}} \frac{1}{6} \sum_{p=1}^d a_p \mathbf{v}_p(\alpha), \quad a_p \in \{-2, -1, 0, 1, 2\}, \quad 1.21.$$

gdzie m – oznacza krok iteracji (przyjmuje wartości całkowite zaczynające się od wartości $m=1$; wektor \mathbf{n}_0 – wyznacza geometryczny środek komórki elementarnej (wzory 1.16.a – 1.16.d), wektor $\mathbf{v}_p(\alpha)$ odpowiada odpowiednio ($p=1, 2, 3, 4$) wektorom z równania (1.16.a – 1.16.d). Funkcja (1.19) zwraca listę złożoną z trzech wartości: kąta α , minimalnej odległości między wybranym punktem, wskazywanym przez wektor \mathbf{n} , a wierzchołkami komórki elementarnej oraz wektor \mathbf{n} . Program wyznaczający promień pokrycia:

```

01  list = {};
02  Timing[Module[{ww, a, i, j, k, q, sa, sb, dd, list0, m},
03  For[a = 0.1, a <= pi/2,
04  (list0 = ss[a, 0, 0, 0, 0, (k1[a] + k2[a] + k3[a] + k4[a])/2];
05  sa = -1/3; sb = 1/3; dd = 1/6; list1 = {0, 0, 0};
06  For[m = 1, m <= mm,
07  (For[i = sa, i <= sb,
08  For[j = sa, j <= sb,
09  For[k = sa, k <= sb,
10  For[q = sa, q <= sb,
11  (ww = ss[a, i, j, k, q, list0[[3]]];
12  If[ww[[2]] > list1[[2]], list1 = ww, 1)];

```

```

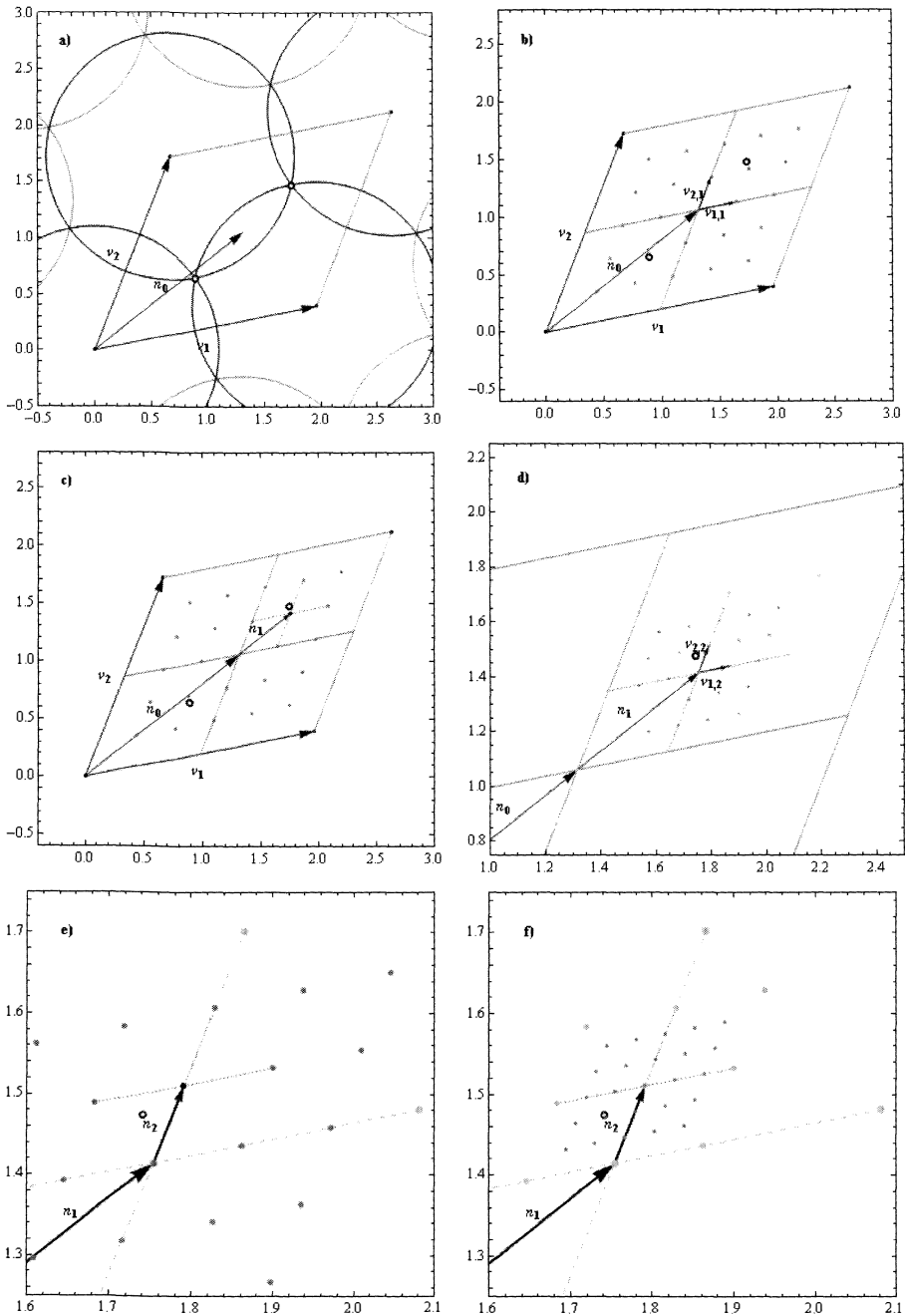
13      q += dd];
14      k += dd];
15      j += dd];
16      i += dd];
17      list0 = list1;
18      sa *= 1/3; sb *= 1/3; dd *= 1/3);
19      m++;]
20      list = Append[list, list0]);
21      a += dda];]]

```

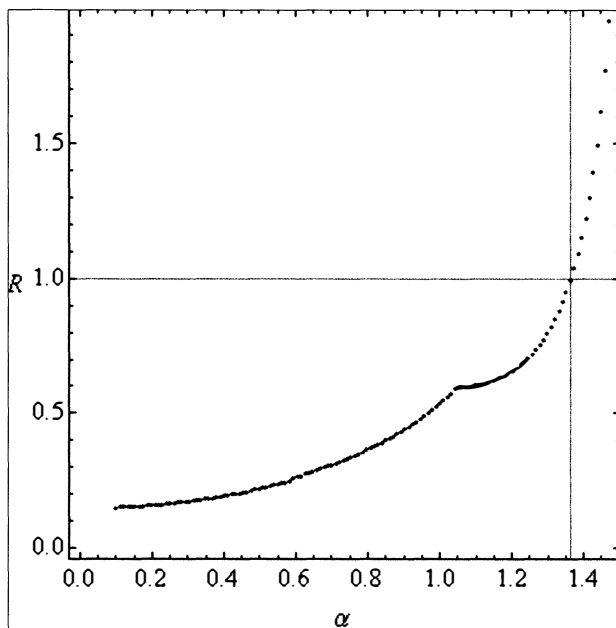
1.22.

W powyższym programie (wydruk 1.22) najbardziej zewnętrzną pętlą jest pętla po parametrze odpowiadającym za wybór wartości kąta α . Wartości kąta zmieniają się od 0.1 do $\pi/2$ z krokiem dda. Dla ustalonej wartości kąta α cztery wewnętrzne pętle po „i”, „j”, „k”, „q” odpowiadają za próbkowanie wnętrza komórki elementarnej (tzn. za wybór punktu, wskazywanego przez wektor \mathbf{n}). Jeśli w wyniku próbkowania zostanie znaleziony punkt, którego odległość do najbliższego wierzchołka jest większa niż dla poprzednio znalezionego (tzn. dla innych wartości parametrów „i”, „j”, „k”, „q”), to taki nowy punkt (w postaci wektora \mathbf{n}) jest zapisywany w liście tymczasowej list1 (wiersz 12). Lista list1 przechowuje ponadto jako dwa pierwsze parametry kąt α oraz minimalną odległość między końcem wektora \mathbf{n} a najbliższym wierzchołkiem komórki elementarnej. Lista list1 (początkowo zainicjowana zerami) przechowywana jest następnie w liście list0 używanej w kolejnej iteracji. Początkowymi parametrami listy list0 (patrz wiersz 4-ty) są: początkowa wartość kąta, cztery początkowe wartości odpowiedzialne za próbkowanie wnętrza komórki elementarnej oraz wektor stanowiący jej geometryczny środek (patrz rysunek Rys 1. a). W pierwszej iteracji ($m=1$) parametry „i”, „j”, „k”, „q” przyjmują wartości z zakresu od $sa=-1/3$ do $sb=1/3$ z krokiem $dd=1/6$. W kolejnych iteracjach parametry sa oraz sb są mnożone przez $1/3$ (linijka 8). Zapewnia to coraz gęstsze próbkowanie wnętrza komórki elementarnej (patrz wzór (1.21); Rys. 1 b) oraz następne). Wyszukiwanie promienia pokrycia dla ustalonego kąta przerywane jest w momencie gdy parametr odpowiedzialny za iterację osiągnie pewną wcześniej ustaloną wartość całkowitą (wiersz 6). Przed przystąpieniem do znalezienia promienia pokrycia dla kolejnej wartości kąta α wynik dołączany jest na koniec listy list (przechowującej wszystkie rozwiązania – wiersz 20; lista ta jest zadeklarowana na samym początku programu). Wynik działania programu dla przykładowej wartości długości wektora wiąza $\Delta\omega'_0 = 1/19235$ został przedstawiony na Rys. 2. Wartość kąta α dla którego promień pokrycia wynosi $R = 1$ można wyznaczyć stosując złożenie algorytmu znajdowania promienia pokrycia z algorytmem znajdowania miejsca zerowego funkcji monotonicznej metodą bisekcji. Dla przykładowej wartości długości wektora wiąza $\Delta\omega'_0 = 1/19235$ optymalna wartość kąta α wynosi:

$$\alpha_{s_2} = 1.36531 \qquad \qquad \qquad 1.23.$$



Rys. 1. Algorytm znajdowania promienia pokrycia dla sieci o zadanych wektorach bazowych. Panele a) – d) stanowią ilustrację działania algorytmu dla przypadku 2-wymiarowej sieci.



Rys. 2. Minimalny promień hipersfery R (oś pionowa) w funkcji kąta α (oś pozioma). Wykres przedstawia wyniki dla wektora wiąza o długości $\Delta\omega'_0 = 1.19235$.

Wartość promienia ($R=1$) dla którego szukamy kąta α wynika z założenia, że oryginalną przestrzeń parametrów sygnału o poziomicach funkcji autokowariancji będącymi hiperelipsoidami przekształcamy w przestrzeń o poziomicach będącymi hipersferami o promieniu zawsze równym jeden. Wektory bazowe oryginalnej przestrzeni sygnału otrzymujemy działając odwzorowaniem odwrotnym na wektory bazowe z przestrzeni z hipersferami.

Wektory bazowe sieci S2 wybieramy w taki sposób, aby dla znalezionej wartości optymalnego kąta pierwszy wektor był sumą wszystkich czterech wektorów krawędziowych (1.16.a – 1.16.d), pozostałe trzy wektory bazowe wybieramy jako identyczne z wektorami (1.16.b – 1.16.d). Otrzymujemy w ten sposób (ze względu na symetrię wektorów krawędziowych względem osi ω_0) macierz wektorów bazowych w której pierwszy wektor bazowy spełnia wiąz ($\Delta\omega'_0 = 1.19235$) wynikający ze stosowania algorytmu szybkiej transformaty Fouriera przy obliczaniu F-statystyk. Macierz generująca sieć S2 dla ($\Delta\omega'_0 = 1.19235$) ma następującą postać:

$$C_2 = \begin{pmatrix} 1.19235 & 0 & 0 & 0 \\ 0.29808 & 1.43017 & 0 & 0 \\ 0.29808 & -0.476722 & 1.34837 & 0 \\ 0.29808 & -0.476722 & -0.674186 & 1.16772 \end{pmatrix} \quad 1.24.$$

3. Wnioski

Algorytm znajdowania promienia pokrycia sieci hipersferami o minimalnym promieniu pozwolił przy złożeniu długości wektora wiąza $\Delta\omega'_0 = 1.19235$ na otrzymanie sieci S2 o gęstości pokrycia różniącej się od gęstości pokrycia sieci optymalne $J_{A_4}^*$ o niecałe 4%. Jest to wynik lepszy (patrz praca [5]) od istniejących sieci (dla $\Delta\omega'_0 = 1.19235$) o ponad 12%. Uzyskany wynik pozwoli skrócić czas przeszukiwania ustalonego obszaru przestrzeni parametrów sygnału przynajmniej o 12%. Gęstość pokrycia sieci optymalnej A_4^* :

$$\rho_{A_4^*} \cong 1.765529, \quad 1.25.$$

Podstawiając macierz generującą (1.24) do równania (1.3) otrzymujemy gęstość pokrycia sieci S2:

$$\rho_{S2} = \frac{\pi^2}{2|\det C_2(\alpha_{S2})|} \cong 1.83792. \quad 1.26.$$

Otrzymana sieć S2 (w przestrzeni z hipersferami) nie zależy od efemeryd (położenia detektora względem środka masy Układu Słonecznego). W związku z tym dla ustalonej długości wektora wiąza wystarczy obliczyć wektory bazowe sieci tylko raz. Wektory bazowe oryginalnej przestrzeni filtrów można dzięki temu otrzymać bardzo szybko działając odwzorowaniem odwrotnym (zależnym od efemeryd) na raz obliczone wektory bazowe sieci S2 w przestrzeni z hipersferami.

Publikacja powstała w ramach realizacji Podlaskiego Funduszu Stypendialnego.

Bibliografia

1. Jaranowski P., A. Królak, B. F. Schutz, *Data analysis of gravitational-wave signals from spinning neutron stars: The signal and its detection*, Physical Review D **58**, 063001 (1998).
2. Jaranowski P., A. Królak, *Data analysis of gravitational-wave signals from spinning neutron stars. II. Accuracy of estimation of parameters*, Physical Review D **59**, 063003 (1991).
3. Jaranowski P., A. Królak, *Data analysis of gravitational-wave signals from spinning neutron stars. III. Detection statistics and computational requirements*, Physical Review D **61**, 062001 (2000).
4. Astone P., Borkowski J., Jaranowski P., Królak A., *Data analysis of gravitational-wave signals from spinning neutron stars. IV. An all-sky search*, Physical Review D **65**, 042003 (2002).
5. Astone P., Borkowski J., Jaranowski P., Pietka M., Królak A., *Data analysis of gravitational-wave signals from spinning neutron stars. V. A narrow-band all-sky search*, Physical Review D **82**, 022005 (2010).