Odboji kroglice na diskretni simetrični sodi verižnici

Anže Kolar (63150020)

15. junij 2018

1 Uvod

Diskretna verižnica predstavlja krivuljo, ki je vpeta na obeh krajiščih, na njene med seboj povezane toge in gibko vpete člene pa deluje le gravitacijska sila. Konstruirati želimo diskretno simetrično verižnico s sodim številom členov ter na njej simulirati odbijanje lahke kroglice.

2 Izračun oblike verižnice

Temelje za izračun je postavil že prof. Zakrajšek v svojem članku o obeh oblikah verižnice [1]. Recimo, da želimo določiti obliko diskretne verižnice z $n+1$ palicami, ki imajo pripadajoče dolžine L_i in mase M_i. Pri določanju lokacije posameznih stičnih točk (x_i, y_i) med členi želimo minimizirati potencialno energijo sistema $F(\mathbf{x}, \mathbf{y}) = \sum_{i=1}^{n+1} M_i \frac{y_i + y_{i-1}}{2}$ s pogoji $(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 = L^2, i \in [n+1]$. To storimo z uvedbo Lagrangevih multiplikatorjev, in s tem poiščemo kandidate za nevezan ekstrem

$$G(\mathbf{x}, \mathbf{y}, \lambda) = \sum_{i=1}^{n+1} \left[M_i \frac{y_i + y_{i-1}}{2} + \lambda_i \left[(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2 - L^2 \right] \right].$$

Z odvajanjem po posameznih spremenljivkah dobimo pogoje

$$\lambda_i \xi_i = \lambda_{i+1} \xi_{x+1}, \tag{1}$$

$$\lambda_i \eta_i - \lambda_{i+1} \eta_{x+1} = -\frac{\mu_i}{2}, \tag{2}$$

$$\xi_i^2 + \eta_i^2 = L_i^2, \tag{3}$$

kjer zaradi lažje pisave označimo z $\xi_i = x_i - x_{i-1}$, $\eta_i = y_i - y_{i-1}$, $i = 1, \ldots, n+1$ ter z $\mu_i = -\frac{M_i + M_{i+1}}{2}, i = 1, \ldots, n$.

Iz (1) sledi $\lambda_i \xi_i = \frac{-1}{2u}$, kjer je u neznana konstanta. Ko ta rezultat vstavimo v (2) in razširimo člene, dobimo $\frac{\eta_i}{\xi_i} = v - u \sum_{j=1}^{i-1} \mu_j, i = 1, \ldots, n+1$, kjer je $v = \frac{\eta_1}{\xi_1}$ še nedoločena konstanta.

Za ξ_i dobimo enačbo

$$\xi_i = \frac{L_i}{\sqrt{1 + \left(v - u\sum_{j=1}^{i-1}\mu_j\right)^2}}, i = 1, \ldots, n+1, \tag{4}$$

ter za η_i enačbo

$$\eta_i = \xi_i\left(v - u\sum_{j=1}^{i-1}\mu_j\right) i = 1, \ldots, n+1. \tag{5}$$

Recimo, da je podana verižnica simetrična z $2k$ členi, iz česar sledi

$$L_i = L_{2k-i+1} \tag{6}$$
$$M_i = M_{2k-i+1} \tag{7}$$
$$\xi_i = \xi_{2k-i+1} \tag{8}$$
$$\eta_i = -\eta_{2k-i+1} \tag{9}$$

Brez škode za splošnost lahko privzamemo tudi, da je namesto z obesišči verižnica podana z razmikom med njima – označimo ga z D –, krajišči pa potem postavimo v točki $(-D/2, 0)$ ter $(D/2, 0)$.

Iz enačb (9) in (5) sledi

$$\xi_i\left(v - u\sum_{j=1}^{i-1}\mu_j\right) = -\xi_{2k-i+1}\left(v - u\sum_{j=1}^{2k-i}\mu_j\right), i = 1, \ldots, k \tag{10}$$

Uporabimo (8) ter izrazimo v. Dobimo

$$v = \frac{1}{2}u\left[2\sum_{j=1}^{i-1}\mu_j + \sum_{j=i}^{2k-i}\mu_j\right]. \tag{11}$$

Zadnji rezultat vstavimo v (4). Po nekaj računanja dobimo, da

$$\xi_i = \frac{L_i}{\sqrt{1 + \left(\frac{1}{2}u\left(2\sum_{j=i}^{k-1}\mu_j + \mu_k\right)\right)^2}}. \tag{12}$$

Spremenljivko u določimo z iskanjem ničle poenostavljene formule iz Zakrajškovega članka,

$$U(u) = \sum_{i=1}^{2k}\xi_i - D = 2\sum_{i=1}^{k}\xi_i - D. \tag{13}$$

Iz dobljenega u lahko iz enačbe (12) določimo spremembe ξ_i ter nato s trikotniškim pravilom še razlike $\eta_i = \sqrt{L_i^2 - \xi_i^2}$.

Točke verižnice $(x_i, y_i), i = 0, \ldots, 2k$, določimo kot $x_i = -D/2 + \sum_{j=1}^{i} \xi_i$ oziroma $y_i = \sum_{j=1}^{i} \eta_i$.

Algoritem za izračun konstante lokacij spojev je implementiran v datoteki `solve_catenary.m`, funkcija za izris pa v `plot_catenary.m`.

3 Izračun odbojev kroglice

Recimo, da kroglica ne pade, ampak jo *izstrelimo* tako, da zadane prvi členek pod kotom ϕ (od navpičnice v smeri urinega kazalca) s hitrostjo v_0 ter na točki, ki je za $r \cdot L_1$ oddaljena od levega obesišča. S simulacijo želimo ugotoviti, kje se bo kroglica nahajala po n odbojih.

Izračun lokacije je iterativen proces, zato moramo najprej opisati, kako obravnavamo posamezni odboj.

3.1 Odboj

Recimo, da kroglica s hitrostjo v pod kotom α od navpičnice pade na dvodimenzionalno ploskev, katere smer opišemo z vektorjem $\mathbf{p} \in \mathbb{R}^2$. Izračun vektorja hitrosti po odboju opravimo v dveh korakih:

1. Hitrost $v \in \mathbb{R}$ razstavimo na dvokomponentni vektor $\mathbf{v}$ v smeri osi (x, y). Velja $\mathbf{v} = [v \cos \alpha, v \sin \alpha]$.
2. Določimo normiran vektor normale ravnine

$$\mathbf{p}_\perp = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \frac{\mathbf{p}}{||\mathbf{p}||}.$$

3. Vektor $\mathbf{v}$ prezrcalimo preko normale $\mathbf{p}_\perp$ [2] z (na primer) Householderjevim zrcaljenjem [3]. Nov vektor hitrosti se glasi

$$\tilde{\mathbf{v}} = \mathbf{v} - 2\mathbf{p}_\perp \left(\mathbf{p}_\perp^T \cdot \mathbf{v} \right).$$

Opisani algoritem je implementiran v datoteki `calc_reflection_angle.m`.

3.2 Izračun naslednjega odboja

Po izračunani smeri odboja nas seveda zanima, kje bo kroglica naslednjič zadela verižnico.

Vemo že, da ima verižnica na svojem najširšem delu širino D. Poznamo tudi hitrost odboja $\mathbf{v} = [v_x, v_y]$ kroglice ter lokacijo (x_0, y_0) le-tega. Iz teh podatkov lahko izračunamo, da bo kroglica steno zadela najkasneje čez D/v_x sekund (zaradi vpliva gravitacijske sile meja ni tesna, spodnja meja pa je logično $t = 0$).

Označimo z $x(t)$ in $y(t)$ obe koordinati kroglice v času t, z $\bar{y}(x)$ pa višino verižnice z abciso x. Z bisekcijo[1] želimo poiskati tak čas t_0, da velja $\bar{y}(x(t)) = y(t)$, torej da kroglica ravno zadane tisti členek verige.

Pozicijo kroglice v času t lahko preprosto izračunamo z nekaj srednješolske matematike[2]:

$$x(t) = x_0 + t \cdot v_x,$$
$$y(t) = y_0 + t \cdot v_y - \frac{g \cdot t^2}{2}.$$

Za dokončni izračun nam manjka le še višina verižnice $\bar{y}(x)$. To izračunamo tako, da v množici členkov palic $\{x_i\}_{i=0}^{2k}$ poiščemo največji (oziroma najmanjši, odvisno od smeri gibanja) tak odmik x_j, da se kroglica nahaja za (oziroma pred) njegovim začetkom. Formalno

$$j = \arg\max_i \{x_i : x_i < x(t)\}.$$

Glede na dobljeni j lahko točko $\bar{y}(x)$ izračunamo kot

$$\bar{y}(x) = y_j + \frac{x(t) - x_j}{x_{j+1} - x_j} \left(y_{j+1} - y_j \right).$$

Paziti je potrebno tudi, da algoritem opozori, če se kroglica v času t nahaja izven meja verižnice. Če ta čas še ni dovolj natančno določen, popravimo zgornjo mejo delilnega intervala, dokler se kroglica ne vrne v meje verižnice. Če pa smo interval že zmanjšali na najmanjšo možno magnitudo, razdalja med kroglico in krivuljo pod njo pa je še vedno neničelna, pa to pomeni, da se je krogla odbila izven meja.

Programska implementacija zgornjega algoritma je na voljo v datoteki `find_impact_time.m` ter `find_link_at.m`.

3.3 Združeni postopek

Sedaj, ko znamo izračunati kot odboja ter čas naslednjega odboja, lahko ta dva algoritma povežemo v celoto. Recimo, da nas zanima pozicija kroglice ob n-tem odboju. Poznamo začetni vektor hitrosti $\mathbf{v}$, vpadni kot ϕ ter točko odboja $\mathbf{s}$. Ponavljamo naslednji postopek:

[1] Med tem ko sicer lahko izračunamo tudi odvod $y'(t) = v_y(t) = v_y(0) + a \cdot t$ in je zaradi tega Newtonova metoda primernejša, pa ne zagotavlja konvergence. En od primerov, ko metoda lahko odpove, je naslednji: recimo, da se kroglica odbije od členka navzgor (torej $y'(t) > 0$) in pade na naslednji člen s hitrostjo $y'(t^*) < 0$. Z Newtonovo metodo iščemo ničle parabole, ki ji odštevamo še zlepek višin členkov (ki v spojih ni odvedljiv). Funkcija ima dve ničli, eno pri $t_0 = 0$, drugo pa v t^*. Tangentna metoda ne nudi garancije, da bo vrnjena ničla t^* – odvisno od začetnega približka in sledečih korakov iteracije, se lahko zgodi tudi, da metoda skonvergira k t_0.

[2] Diskretiziranje in reševanje s parcialnimi enačbami (npr. preko funkcije ode45) je seveda tudi možno, a nekoliko manj intuitivno.

1. določimo obliko verižnice s točkami $\{(x_i, y_i)\}_i$.
2. za $i = 1, \ldots, n$:
   (a) če se kroglica nahaja izven meja verižnice: končamo
   (b) če $i = n$: končamo
   (c) če $i < n$: Določimo člen pod kroglico ter izračunamo smer odboja v točki $\mathbf{s}$ glede na vektor hitrosti $\mathbf{v}$ ter kot ϕ. Z bisekcijo poiščemo novo točko odboja $\tilde{\mathbf{s}}$ ter nadaljujemo iteracijo s tisto točko.
3. Če velja $i = n$, je zadnja izračunana pozicija enaka lokaciji po n odbojih.

Opisan postopek je implementiran v datoteki `pos_after_n_hits.m`. Skupaj z grafičnim prikazom je na voljo tudi v datoteki `ball_dynamics.m`.

4 Rezultati

Sliki 1 in 2 prikazujeta odboje kroglic za različne konfiguracije verižnice ter začetne parametre meta.

Iz podatkov v tabeli 1 in grafičnega prikaza istih podatkov na sliki 3 lahko razberemo, da čas izvajanja linearno narašča s številom odbojev.

Poročilu so dodane naslednje (glavne) skripte:

catenary izračuna obliko verižnice glede na parametre D, $\mathbf{L}$ in $\mathbf{M}$.

position_after_n_hits izračuna lokacijo kroglice po n odbojih glede na podano verižnico $\mathcal{X}$, začetni odmik r, kot α in hitrost v_0.

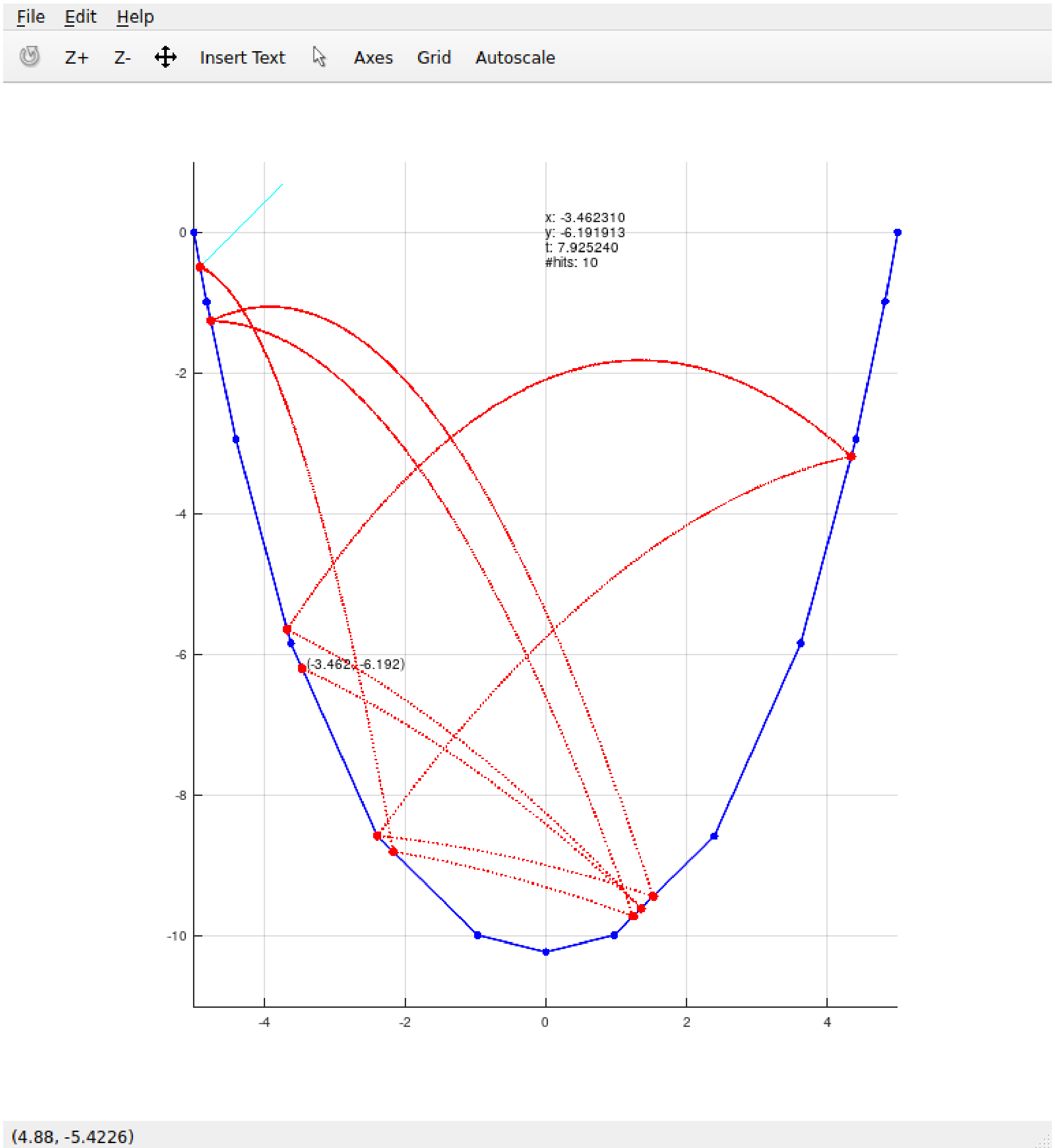
run_all je konzolni vmesnik, ki omogoča enostaven vnos parametrov za zgornji dve skripti.

Literatura

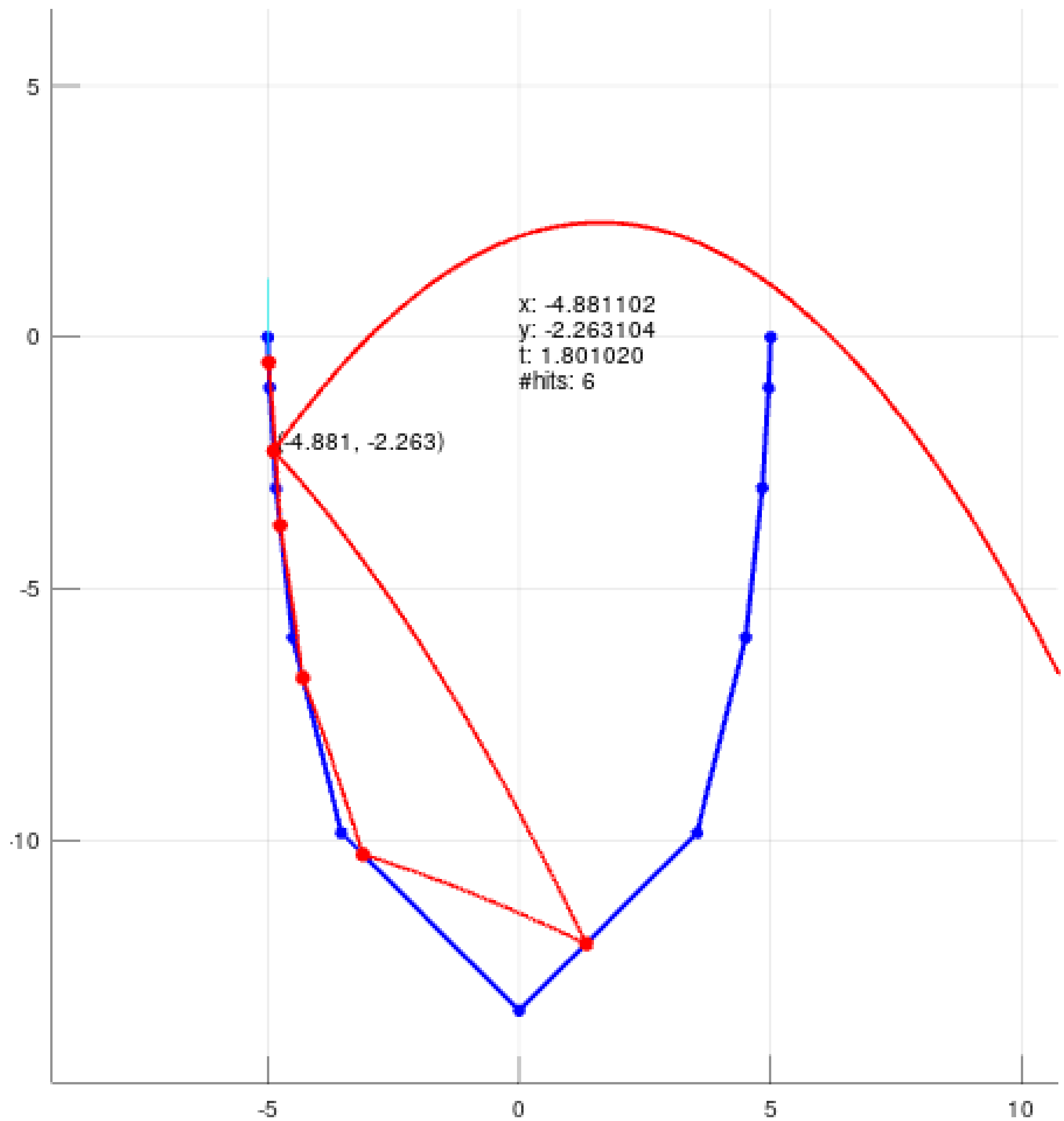
[1] E. Zakrajšek, “Verižnica”, 1999.

[2] *Specular reflection*, Dosegljivo: `https://en.wikipedia.org/wiki/Specular_reflection`, Dostopano: 23. 5. 2018.

[3] A. Householder, “Unitary Triangularization of a Nonsymmetric Matrix”, 1958.



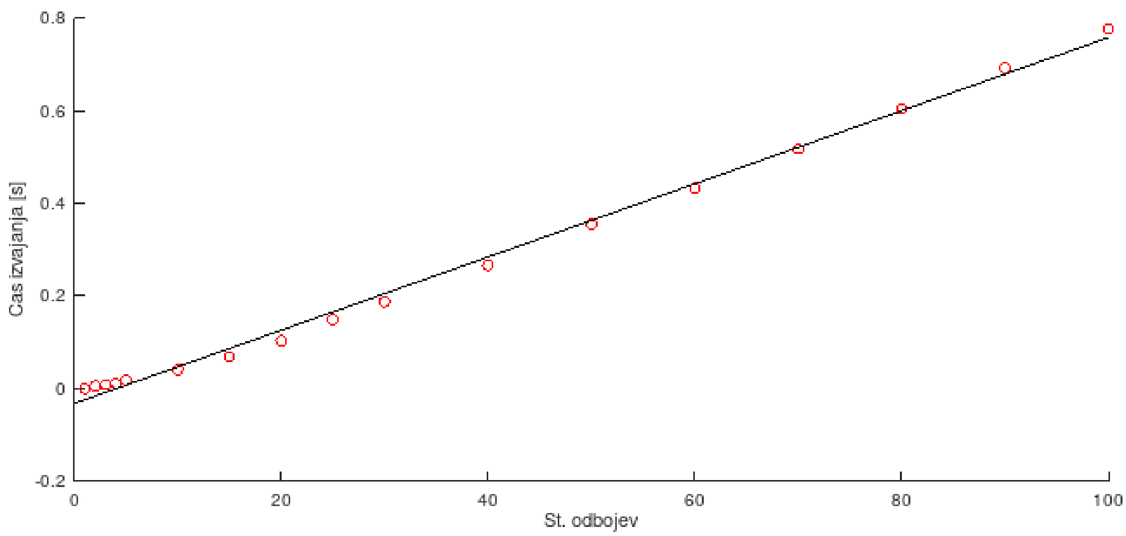
Slika 1: Padec kroglice na verižnico z razmakom $D = 10$, dolžinami (leve strani) palic $[1, 2, 3, 3, 2, 1]$ ter masami $[2, 1, 3, 3, 2, 1]$. Kroglica pade na 50 % prve palice pod kotom $\frac{\pi}{4}$ ter s hitrostjo $v_0 = 2.5$ in se odbije desetkrat.



Slika 2: Padec kroglice na verižnico z razmakom $D = 10$, dolžinami (leve strani) palic $[1, 2, 3, 4, 5]$ ter masami $[5, 4, 3, 2, 1]$. Kroglica pade na 50 % prve palice s hitrostjo $v_0 = 10$ pod kotom 0 in se po šestih odbojih odbije izven meja verižnice.

Tabela 1: Čas izvajanja programa pri spustu kroglice na 50 % prvega členka s hitrostjo $v_0 = 0.5$ pod kotom 0 pri različnem številu odbojev. Verižnica ima 20 enakih členov teže $M_i = 1$ z dolžinami $L_i = 1$ in je razmaknjena za $D = 15$. Podatki so povprečeni nad petimi ponovitvami.

Št. odbojev	Čas izvajanja [s]
1	$2.09 \cdot 10^{-4}$
2	$6.32 \cdot 10^{-3}$
3	$8.73 \cdot 10^{-3}$
4	$1.21 \cdot 10^{-2}$
5	$1.80 \cdot 10^{-2}$
10	$4.15 \cdot 10^{-2}$
15	$6.93 \cdot 10^{-2}$
20	$1.03 \cdot 10^{-1}$
25	$1.49 \cdot 10^{-1}$
30	$1.88 \cdot 10^{-1}$
40	$2.66 \cdot 10^{-1}$
50	$3.54 \cdot 10^{-1}$
60	$4.33 \cdot 10^{-1}$
70	$5.18 \cdot 10^{-1}$
80	$6.04 \cdot 10^{-1}$
90	$6.92 \cdot 10^{-1}$
100	$7.77 \cdot 10^{-1}$



Slika 3: Vizualizacija podatkov iz tabele 1, torej časa izvajanja v odvisnosti od števila odbojev, z dodanim najbolje prilegajočim polinomom (stopnje 1).