

## Penerapan Thresholding Bertahap Pada Proses Denoising Sinyal Berbasis Wavelet

Mahmud Yunus

Jurusan Matematika  
Institut Teknologi Sepuluh Nopember, Surabaya  
yunusm@matematika.its.ac.id

### Abstrak

Metode-metode berbasis wavelet menjadi piranti yang sangat efisien dan semakin menarik untuk diteliti. Dalam makalah ini disampaikan prosedur penerapan thresholding secara bertahap (TSB) pada masalah denoising sinyal. Sebagai contoh kasus, TSB akan diterapkan untuk denoising sinyal (data) satu dimensi.

TSB diterapkan pada sinyal yang diberikan, dengan terlebih dahulu menentukan nilai threshold yang paling sesuai untuk tiap tahap dekomposisi, kemudian proses denoising dilakukan serentak dengan proses rekonstruksi. Hasil penerapan TSB akan dibandingkan dengan dua metode yang telah cukup dikenal. Pertama, cara Donoho dan Johnstone (1995) yang menerapkan threshold global  $\lambda \sim \sigma\sqrt{2\log n}$  untuk  $j > j_0$ , dengan  $j_0$  resolusi terakhir yang dipilih. Kedua, dibandingkan dengan metode berbasis FDR (False discovery error rate) yang dikemukakan oleh Abramovich dan Benjamini (1998).

**Kata kunci:** *denoising, thresholding bertahap, wavelet*

## 1 Pendahuluan

Misal diberikan data

$$y_i = x(t_i) + \varepsilon_i, \quad i = 1, \dots, n \quad (1)$$

dengan  $t_i = i/n$ , dan  $\varepsilon_i$  noise normal i.i.d. dengan mean nol dan variansi  $\sigma^2$ . Selanjutnya akan diduga nilai fungsi  $x(\cdot)$  yang tidak diketahui dari data yang diberikan tersebut dan tanpa adanya asumsi apapun mengenai bentuk  $x$ . Saat ini, cara-cara yang menggunakan wavelet menjadi semakin menarik dan merupakan piranti yang efisien dalam estimasi fungsi sejenis itu.

Dalam penderetan dengan deret Fourier basis yang digunakan adalah fungsi-fungsi sinus dan cosinus dalam berbagai frekuensi, sedangkan dengan deret wavelet digunakan fungsi-fungsi basis  $\psi_{jk}$ , yang merupakan hasil

translasi dan dilasi fungsi tunggal  $\psi$  yang disebut *wavelet induk*, yaitu  $\psi_{jk} = 2^{j/2}\psi(2^j t - k)$   $j, k \in \mathbb{Z}$ . Fungsi  $y$  di atas didekomposisi menggunakan fungsi-fungsi basis ortonormal  $\psi_k$ . Dari satu wavelet induk  $\psi$ , fungsi  $\psi_{jk}$  yang berkaitan membentuk basis ortonormal di  $L^2(\mathbb{R})$  (contoh-contoh wavelet induk yang demikian dapat dilihat di buku Daubechies [3]). Dalam hal ini, fungsi  $x$  dalam (1)

$$x(t) = \sum_j \sum_k d_{jk} \psi_{jk}$$

dengan koefisien-koefisien wavelet

$$\begin{aligned} d_{jk} &= \langle y, \psi_{jk} \rangle = \int y(t) \psi_{jk}(t) dt \\ &= 2^{j/2} \int y(t) \psi(2^j t - k) dt \end{aligned}$$

Berlainan dengan sinus dan cosinus, wavelet

terlokalisir sekaligus dalam frekuensi (melalui dilasi) dan waktu (melalui translasi). Sifat lokal ini merupakan salah satu sifat istimewa wavelet, sebab hanya diperlukan sedikit fungsi basis untuk menyajikan suatu fungsi pada himpunan resolusi tertentu.

Misal data diberikan di titik-titik  $t_i = i/n$  dengan ukuran sampel  $n = 2^{J+1}$  untuk suatu bilangan bulat positif  $J$ . Tanpa kesertaan noise acak pada data, dapat ditemukan  $m = n - 1$  koefisien wavelet  $d_{jk}$  untuk fungsi  $x$ ,  $j = 0, \dots, J$ ,  $k = 0, \dots, 2^j - 1$ , dengan menerapkan transformasi wavelet diskrit (TWD) dari vektor tanpa noise  $\mathbf{x} = (x(t_1), \dots, x(t_n))'$ . Algoritma TWD yang dapat digunakan untuk memperoleh  $d_{jk}$  telah dirumuskan oleh Mallat [6]. Karena dengan hanya sedikit fungsi basis wavelet sudah mencukupi untuk menyajikan suatu sinyal, maka sangat beralasan apabila diasumsikan bahwa hanya sedikit  $d_{jk}$  yang memuat informasi tentang sinyal yang sebenarnya, sedangkan yang lainnya muncul akibat adanya noise acak. Untuk hal ini, ekstraksi dapat dilakukan dengan proses thresholding pada koefisien-koefisien  $d_{jk}$ :

$$d_{jk}^* = \begin{cases} d_{jk}, & |d_{jk}| \geq \lambda, \\ 0, & |d_{jk}| < \lambda. \end{cases}$$

dengan  $\lambda$  nilai threshold yang dipilih. Perlu dicatat bahwa kebanyakan kasus denoising dalam praktek, orang yang berbeda akan berbeda pula memilih nilai threshold. Hal ini dapat dimungkinkan dari pengamatan dan pengalaman pemakai yang memandang bagian mana dari data yang dapat dikorbankan.

Dalam makalahnya, Donoho dan Johnstone [4] menggunakan threshold global  $\lambda \sim \sigma\sqrt{2 \log n}$  untuk semua  $j$  yang lebih besar dari suatu  $j_0$ . Dalam prakteknya,  $\sigma$  selalu tidak diketahui dan biasanya diduga dari koefisien-koefisien resolusi paling halus. Donoho dan Johnstone menyarankan untuk selalu melibatkan koefisien-koefisien pada tahap  $j_0$  yang pertama, walaupun koefisien-koefisien tersebut tidak lolos tahap thresholding. Dalam makalah tersebut digunakan  $j_0 = 5$ . Secara intuitif, pemilihan  $j_0$  yang tepat seharusnya bergantung pada kemulusan dari fungsi yang diduga dan pada tingkat noise yang ada.

Abramovich dan Benjamini [1] memperkenalkan cara menentukan nilai threshold yang didasarkan pada prosedur FDR (False Discovery error Rate) yang kemudian dinamainya

dengan FDRC (False Discovery Rate of Coefficients). FDRC menunjukkan perbaikan dibandingkan dengan metode yang menerapkan threshold global tetap, yang didasarkan pada perbandingan Relative Mean Square Error untuk berbagai fungsi-uji (sinyal-uji) dengan berbagai tingkat noisennya.

Secara singkat prosedur FDRC adalah sebagai berikut:

- (i) Untuk setiap  $\hat{d}_{jk}$  hitung nilai- $p$  dua-sisi yang bersesuaian,  $p_{jk}$ , dengan menguji  $H_{jk} : d_{jk} = 0$ ,
- (ii) Urutkan  $p_{jk}$  berdasarkan nilainya,  $p_{(1)} \leq p_{(2)} \leq \dots \leq p_{(m)}$ .
- (iii) Pilih  $i_0$  nilai terbesar  $i$  dimana  $p_{(i)} \leq (i/m)q$ . Untuk  $i_0$  tersebut tentukan  $\lambda_{i_0} = \sigma\Phi^{-1}(1 - p_{(i_0)}/2)$ .
- (iv) Lakukan thresholding untuk semua koefisien pada tahap  $\lambda_{i_0}$ .

Ditunjukkan dalam makalah tersebut, bahwa thresholding yang efektif dapat dikerjakan pada tahap dekomposisi antara  $\lambda_{min} = \sigma\Phi^{-1}(1 - q/2)$  dan  $\lambda_{max} = \sigma\Phi^{-1}(1 - q/2n)$ .

Makalah ini hanya menyampaikan penerapan prosedur TSB pada sinyal (fungsi) pada garis real  $\mathbb{R}$ , sedangkan untuk perluasannya pada  $\mathbb{R}^p$  yang lebih umum, dan khususnya untuk sinyal atau citra di  $\mathbb{R}^2$  dapat diturunkan dengan tahapan yang serupa.

## 2 Prosedur Thresholding Secara Bertahap

Berdasarkan sifat dasar transformasi wavelet, diketahui bahwa tingkat resolusi yang lebih halus (tahap dekomposisi yang lebih tinggi) akan hanya memuat informasi detail dan memberikan sinyal rata-rata dalam resolusi yang lebih kasar. Oleh karena itu, dengan menghapus tingkat resolusi yang paling halus, dan merekonstruksinya kembali akan diperoleh pengaruh penghalusan seperti penerapan filter linear yang sangat elementer. Dalam hal ini noise dapat tereduksi, tetapi kemungkinan juga akan mengakibatkan mengaburnya sinyal asal.

Telah diketahui pula bahwa kebanyakan koefisien wavelet adalah sangat kecil. Akan tetapi, masing-masing tahap dekomposisi menghasilkan koefisien-koefisien yang membawa sifat-sifat yang berbeda sesuai dengan

tingkat resolusinya. Untuk itu, perlakuan pada koefisien-koefisien pada masing-masing tahap dekomposisi perlu dicermati dan seharusnya diperlakukan berbeda.

Prosedur thresholding secara bertahap (TSB) akan diterapkan untuk koefisien-koefisien pada tahap dekomposisi yang berbeda. Dengan asumsi noise  $\varepsilon$  pada (1) merupakan noise acak bebas (i.i.d.)  $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ . Dengan TWD yang mempunyai kompleksitas linear  $O(N)$ , perlakuan yang berbeda untuk tiap-tiap tahap dekomposisi tidak menjadi masalah, setidaknya dalam hal komputasinya.

Dengan TWD, persamaan (1) dapat ditulis kembali sebagai

$$\mathbf{y} = \mathbf{x} + \mathbf{v} \tag{2}$$

dengan  $\mathbf{y}$  memuat TWD dari sinyal yang diamati,  $\mathbf{x}$  vektor TWD dari sinyal tanpa noise (sinyal bersih), dan  $\mathbf{v}$  vektor TWD dari noise. Dalam permasalahan ini akan dicari  $\hat{\mathbf{x}}$  dengan menerapkan thresholding pada koefisien-koefisien detail  $\mathbf{y}_j$  tiap tahap dekomposisi.

Pada dasarnya, penentuan nilai threshold yang diterapkan untuk TSB adalah nilai threshold Donoho dan Johnstone, hanya saja sifat global pada seluruh koefisien detail diganti dengan koefisien detail masing-masing tahap dekomposisi. Sedangkan taksiran nilai  $\sigma$  ditentukan dengan cara Abramovich dan Benjamini, yaitu dengan menduga nilainya sesuai SNR yang diharapkan, yang kemudian nilai tersebut diskala menyesuaikan tingkat noise pada tahap tersebut. Penskalaan didasarkan pada nilai simpangan maksimum absolut dibagi 0.6745, nilai yang cocok untuk penanganan noise acak seperti asumsi yang digunakan di sini. Nilai ini digunakan oleh Hansen dkk. [5].

Secara ringkas, langkah-langkah penerapan prosedur TSB untuk denoising sinyal  $\mathbf{y}$  dapat dirumuskan sebagai berikut:

- (i) Transformasikan  $\mathbf{y}$  dengan TWD sehingga diperoleh koefisien-koefisien TWD:  $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{j_0}$ , dengan  $\mathbf{y}_j$  bersesuaian dengan tahap ke- $j$ .
- (ii) Untuk masing-masing  $\mathbf{y}_j$  tentukan nilai threshold  $\lambda_j$  yang memenuhi  $\|\mathbf{y}_j\|/\sigma = \delta$  dengan  $\delta$  adalah SNR yang diharapkan.
- (iii) Lakukan penskalaan pada masing-masing  $\lambda_j$  dengan menghitung  $\max(|y_{jk}|)/0.6745$ .

- (iv) Kerjakan thresholding pada koefisien-koefisien detail  $\mathbf{y}_j$  dengan nilai threshold  $\lambda_j$  yang bersesuaian.

- (v) Kerjakan invers TWD pada  $\mathbf{y}$  untuk mendapatkan  $\hat{\mathbf{x}}$ .

Langkah ke-(iv) dikerjakan dengan ‘soft’ thresholding [2], yaitu dengan menghitung ‘rata-rata’  $y_{jk} \leftarrow \text{sign}(y_{jk}) * R(s_j)$ , dimana  $R(s_j) = (s_j + |s_j|)/2$  dan  $s_j = |y_{jk}| - \lambda_j$ .

### 3 Simulasi Perhitungan Numerik

Dalam perhitungan numerik, diamati hasil numerik metode Donoho dan Johnstone (DJ) untuk tiga estimator yang berkaitan dengan tiga nilai tahap awal thresholding  $j_0$  yang berbeda, yaitu  $j_0 = 1$  (DJ1),  $j_0 = 3$  (DJ3), dan  $j_0 = 5$  (DJ5). Sedangkan untuk metode FDRC dihitung untuk  $q = 0.01$  (AB01) dan  $q = 0.05$  (AB05) sebagaimana dalam makalah Abramovich dan Benjamini.

Prosedur tersebut di atas, sekaligus TSB, diterapkan pada lima sinyal-uji, SU1, SU2, SU3, SU4, dan SU5, berikut ini (lihat lampiran Gambar Sinyal Uji):

$$SU1 : x(t) = (t - 0.04)^2 \text{ (contoh fungsi kuadrat, [1])}$$

$$SU2 : x(t) = \min(2t, -2(t - 1)) \text{ (contoh fungsi segitiga, [1])}$$

$$SU3 : x(t) = [t - 0.3] - [t - 0.7] \text{ (contoh fungsi kotak, [1])}$$

$$SU4 : x(t) = \sum_j h_j K(t-t_j), \text{ dengan } K(t) = (1 + \text{sign}(t))/2$$

$$(t_j) = (0.1, 0.13, 0.15, 0.23, 0.25, 0.40, 0.44, 0.65, 0.76, 0.78, 0.81),$$

$$(h_j) = (4, -5, 3, -4, 5, -4.2, 2.1, 4.3, -3.1, 2.1, -4.2)$$

(contoh fungsi kotak, [4])

$$SU5 : x(t) = \sum_j h_j K((t - t_j)/w_j), \text{ dengan } K(t) = \max((1 - |t|)^4, 0)$$

$$(t_j) = (0.1, 0.13, 0.15, 0.23, 0.25, 0.40, 0.44, 0.65, 0.76, 0.78, 0.81),$$

$$(h_j) = (4, 5, 3, 4, 5, 4.2, 2.1, 4.3, 3.1, 5.1, 4),$$

$$(w_j) = (0.005, 0.005, 0.006, 0.01, 0.01, 0.03, 0.01, 0.01, 0.005, 0.008, 0.005)$$

(contoh fungsi bumps, [4])

Untuk masing-masing sinyal-uji simulasikan dengan empat nilai  $\sigma$  yang berbeda yang memberikan SNR (signal to noise ratio) berturut-turut  $\delta = 1, 2, 4, 7$ . Data yang digunakan adalah 1024 titik data yang dipilih pada  $[0, 1]$  dengan jarak sama. Perhitungan dilakukan dengan mengambil 100 kali percobaan untuk masing-masing kombinasi sinyal-uji dan  $\sigma$  dengan menambahkan noise acak bebas  $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$  pada  $x(t(i/n))$ .

Untuk mendapatkan vektor TWD  $\hat{\mathbf{x}}$  digunakan algoritma dekomposisi-rekonstruksi dari Mallat [6]. Sedangkan pada tahap dekomposisi, koefisien wavelet  $\hat{d}_{jk}$  diperoleh menggunakan basis wavelet Daubechies  $D_4$  [3]. Sedangkan komputasinya menggunakan Matlab. Ukuran yang digunakan untuk perbandingan adalah *mean square error*  $\mathbf{MSE} = n^{-1} \|\mathbf{x} - \hat{\mathbf{x}}\|_{\ell^2}$  yang diambil dari rata-rata 100 sinyal uji dengan beragam noise acak yang diberikan.

Untuk penerapan TSB digunakan tahap dekomposisi seperti metode DJ, yaitu untuk  $j_0 = 3$  (TSB3),  $j_0 = 5$  (TSB5), dan  $j_0$  tahap maksimal (TSBmax). Untuk tiap sinyal uji, diperoleh TSB ‘terbaik’ di antara ketiganya, yakni yang mencapai nilai MSE minimum. Untuk itu digunakan ‘MSE relatif’ yang dihitung dengan  $(\min \text{MSE}_j) / \text{MSE}_i$ ,  $i = 1, 2, 3$ . Dari Tabel 1 tampak bahwa untuk sinyal dengan noise cukup tinggi TSBmax adalah yang terbaik diterapkan pada sinyal yang cukup mulus (SU1, SU2, SU3), sedangkan untuk sinyal berosilasi (SU4 dan SU5) TSB5 menyajikan hasil yang lebih baik. Untuk sinyal asal yang cukup halus (tingkat noise cukup rendah), ketiga TSB menunjukkan hasil yang sama pada kasus sinyal yang cukup mulus (SU1).

## 4 Penutup

Berdasarkan Tabel 2, tampak bahwa untuk sinyal dengan noise cukup rendah dan tingkat osilasi tinggi, AB05 menunjukkan hasil terbaik diantara delapan cara yang ditampilkan. Sedangkan untuk sinyal yang cukup mulus dengan tingkat noise rendah, ketiga TSB memberikan hasil terbaiknya. Keistimewaan TSBmax ditunjukkan dengan hasil terbaik untuk

sinyal cukup mulus (SU1, SU3, SU3) dengan noise tinggi.

Tampak pula pada Tabel 2, bahwa nilai MSE relatif terkecil (dicetak tebal) tidak muncul pada kolom-kolom TSB. Hal menunjukkan bahwa secara umum, TSB bukan metode yang dapat diabaikan begitu saja. Hasil ini masih secara numerik dan kemungkinan perbaikan prosedur thresholding bertahap seperti TSB ini masih dapat diperbaiki, salah satunya dengan memeriksa secara analitik pemilihan  $\sigma$  yang lebih sesuai.

Akhirnya dapat disimpulkan bahwa, TSB dapat diterapkan dengan hasil baik khususnya untuk sinyal yang cukup mulus bahkan untuk tingkat noise yang beragam. Selain itu, berdasarkan hasil numerik yang disajikan dalam Tabel 2 secara umum TSB memberikan hasil yang cukup baik, yaitu dengan MSE relatif lebih besar dari 0.8.

## Daftar Pustaka

- [1] Abramovich, F., Benjamini, Y., *Adaptive Thresholding of Wavelet Coefficients*, J. Royal Statistics Soc., Ser. B (61), 1998.
- [2] Bultheel, A., *Wavelets with Applications in Signal and Image Processing*, Lecture Notes, Katholieke Universiteit, Leuven, Belgium, 2001.
- [3] Daubechies, I., *Ten Lectures on Wavelets*, SIAM, Philadelphia, 1992.
- [4] Donoho, D.L. and Johnstone, I.M., *Adapting to Unknown Smoothness via Wavelet Shrinkage*, Journal of American Statistics Ass., 90, 1200–1244, 1995.
- [5] Hansen, M., Malfait, M., Bultheel, A., *Generalized Cross Validation for Wavelet Thresholding*, Signal Processing, 56, 33–44, 1997.
- [6] Mallat, S.G., *A Theory for Multiresolution Signal Decomposition: the Wavelets Representation*, IEEE Transac. on Pattern Analysis and Machine Intell., vol 11 (7), pp. 674–693, 1989.

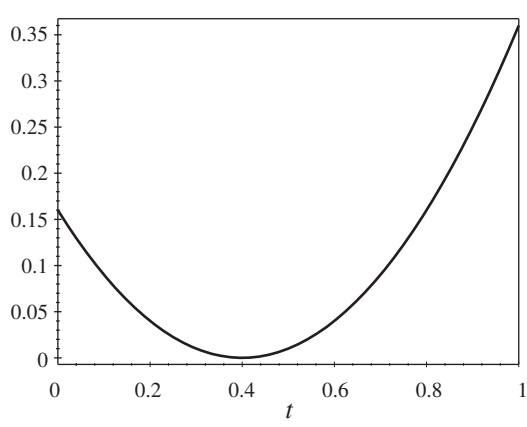
## Lampiran

Tabel 1: MSE Relatif untuk TSB;  $\delta = 4$   
(diambil rata-rata dari 100 uji)

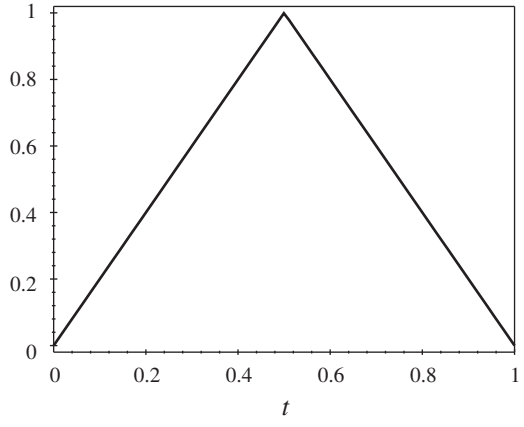
Sinyal	TSB3	TSB5	TSBmax
SU1	1.000	1.000	1.000
SU2	0.962	1.000	0.981
SU3	0.992	1.000	0.993
SU4	0.928	1.000	0.986
SU5	1.000	0.980	0.886

Tabel 2: MSE Relatif (rata-rata dari 100 uji)

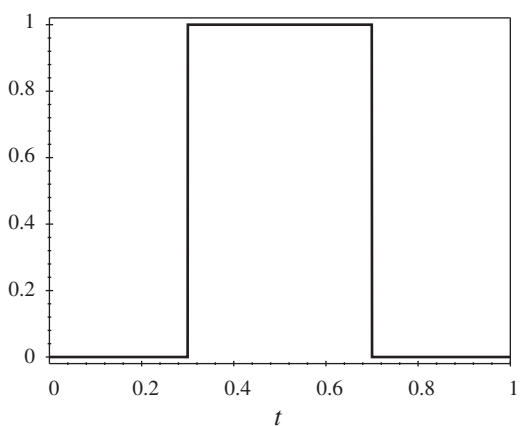
SNR	SU	DJ1	DJ3	DJ5	AB01	AB05	TSB3	TSB5	TSBmax
1	SU1	0.670	0.717	0.470	0.805	0.760	0.911	0.953	1.000
	SU2	0.697	1.000	0.553	0.687	0.686	0.918	0.926	1.000
	SU3	0.718	0.793	0.866	0.694	0.772	0.959	0.862	1.000
	SU4	0.737	0.765	1.000	0.773	0.801	0.951	0.960	0.916
	SU5	0.845	0.878	0.959	0.857	1.000	0.953	0.977	0.847
2	SU1	0.796	1.000	0.527	0.786	0.761	0.923	0.960	0.955
	SU2	0.793	1.000	0.729	0.791	0.767	0.954	0.947	0.971
	SU3	0.770	0.867	1.000	0.752	0.816	0.880	0.880	0.843
	SU4	0.913	0.922	1.000	0.934	0.992	0.954	0.863	0.968
	SU5	0.787	0.793	0.880	0.857	1.000	0.912	0.964	0.899
4	SU1	0.728	1.000	0.676	0.806	0.774	1.000	1.000	1.000
	SU2	0.844	1.000	0.885	0.849	0.799	0.962	1.000	0.981
	SU3	0.857	0.873	0.956	0.875	1.000	0.992	1.000	0.993
	SU4	0.774	0.781	0.849	0.848	1.000	0.928	1.000	0.986
	SU5	0.631	0.631	0.719	0.763	1.000	1.000	0.980	0.886
7	SU1	0.878	0.808	0.690	0.692	0.859	1.000	1.000	1.000
	SU2	0.687	0.788	0.944	0.682	0.677	0.941	1.000	0.842
	SU3	0.899	0.897	0.922	0.910	1.000	0.899	0.979	0.940
	SU4	0.621	0.637	0.713	0.784	1.000	0.999	0.923	0.921
	SU5	0.600	0.600	0.610	0.727	1.000	0.989	0.915	0.901



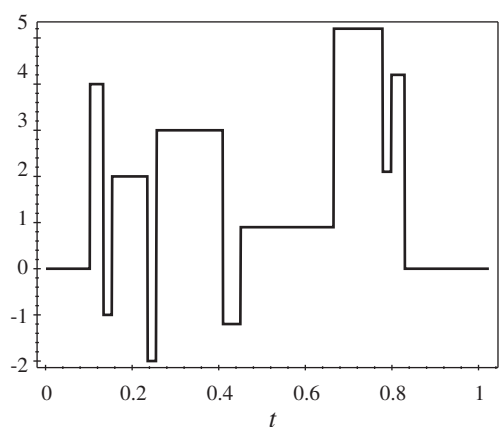
*SU 1*



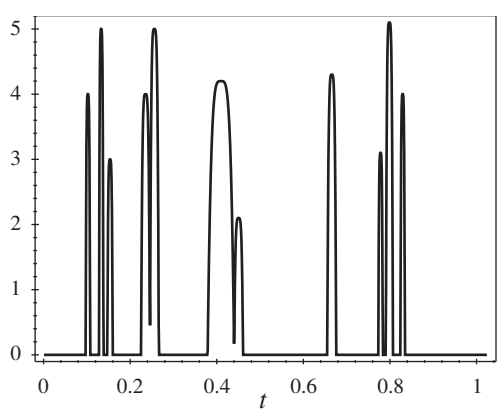
*SU 2*



*SU 3*



*SU 4*



*SU 5*

Gambar Sinyal Uji