ლაბორატორია 1

დისკრეტული დროის სისტემები

1 შესავალი

ყველა სისტემა, რომლის შემავალი და გამომავალი დისკრეტული დროის სიგნალებია დისკრეტული დროის სისტემაა. სისტემის ცნება ძალიან ზოგადია და მრავალი სხვადასხვა ტიპის სისტემის და პროცესის მოდელირებისთვის გამოიყენება: ეს შეიძლება იყოს აუდიო და ვიდეო დამუშავება, ეკონომიკური პროცესები, რადარული/სონარული სისტემები.

უწყვეტი დროის სიგნალების დამუშავებისთვის ელექტრული წრედები გამოიყენება, რომლებიც რეზისტორების, კონდენსატორების და ინდუქტორებისგან შედგება და რომელთა აღწერა დიფერენციალური განტოლებებებით ხდება. წრედში ძაბვების და დენის გასაგებად დიფერენციალური განტოლებების ამოხსნაა საჭირო.

დისკრეტული დროის სიგნალების დასამუშავებლად დისკრეტული დროის სისტემები გამოიყენება. დისკრეტულ დროის სისტემას სხვაობიანი განტოლებით ან/და შესაბამისი ბლოკ-სქემით ან სტრიმ-გრაფით წარმოვადგენთ. განვიხილოთ შემდეგი სხვაობიანი განტოლება.

$$y[n] = y[n-1] - 2x[n] + 3x[n-1] \tag{1}$$

ეს განტოლება დისკრეტული დროის სისტემას აღწერს. შემავალი სიგნალია, $x[n]$ და გამომავალია, $y[n]$. სისტემა შეგვიძლია ბლოკ-სქემითაც წარმოვადგინოთ.

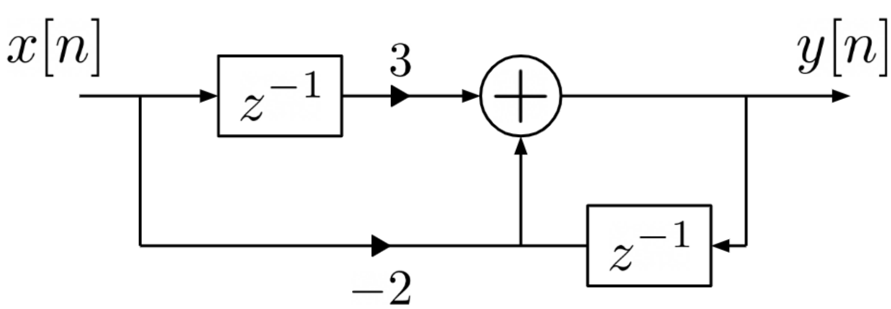

Figure 1: დისკრეტული დროის სისტემის დიაგრამა

ფორმალურად, დისკრეტული დროის სისტემა შეგვიძლია შემდეგნაირად წარმოვადგინოთ, $H\{\cdot\}$, გამომავალი სიგნალი მოცემულია როგორც, $y[n] = H\{x[n]\}$, სადაც $x[n]$ შემავალი მიმდევრობაა. ზოგად, დისკრეტული დროის სისტემაში შესაძლოა, რომ გამომავალი სიგნალის ანათვალი $y[n_0]$ მომავალ მნიშვნელობებზეც, $y[n_0 + \tau]$ იყოს დამოკიდებული, სადაც $\tau > 0$. ასეთი დისკრეტული სისტემის რეალურ დროში დანერგვა შეუძლებელია, თუმცა როდესაც მთლიანი სიგნალი სრულად გვაქვს

მოცემული, მაგალითად სურათების დამუშავებისას, ასეთი სისტემებიც გამოიყენება. დისკრეტული დროის სისტემა, რომლის ახლანდელი გამომავალი მნიშვნელობის გამოსათვლელად მომავალი შემავალი ან გამომავალი მნიშვნელობებია საჭირო არა-კაუზალური სისტემა ეწოდება.

აჩვენეთ, რომ სხვაობიანი გატოლება (1)-ით აღწერილი სისტემა კაუზალური, დროით ინვარიანტული, წრფივი სისტემაა. დისკრეტული სისტემების სრულად გასაანალიზებლად საჭიროა მათი კლასიფიკაციის გააზრება: წრფივი/არა-წრფივი, დროით ინვარიანტული/დროით არა-ინვარიანტული, კაუზალური/არა-კაუზალური და სტაბილური/არა-სტაბილური. უმეტესად, ციფრული სიგნალების დამუშავებისას წრფივ, დროით-ინვარიანტულ, სტაბილურ და კაუზალურ დისკრეტული დროის სისტემებზე ვკონცენტრირდებით. წრფივ, დროით ინვარინტულ სისტემას ინიციალეებით, ***LTI*** (linear time-invariant), ავღიშნავთ.

2 დისკრეტული სიგნალების წარმოდგენა და გრაფიკები MATLAB-ში

დისკრეტული სისტემის დანერგვამდე, გვჭირდება ვიცოდეთ დისკრეტული სიგნალების (მიმდევრობების) წარმოდგენის და მათზე მარტივი ოპერაციების ჩატარების მეთოდები.

MATLAB-ში დისკრეტული დროის სიგნალების გრაფიკის ასაგებად ბრძანებას, *stem*, ვიყენებთ.

ბრძანების ფანჯარაში შეიყვანეთ:

```
n = 0:0.1:10;
y = sin(pi*n);
stem(n,y)
```

კომპიუტერში უწყვეტი სიგნალის წარმოდგენა შეუძლებელია, რადგან ამისთვის უსასრულო მეხსიერებაა საჭირო. უწყვეტი სიგნალის უკეთესი მიახლოებისთვის შეგვიძლია დროით ვექტორში ბიჯის ზომა შევამციროთ. ამ შემთხვევაში *plot* ბრძანება გამოიყენეთ:

```
n = 0:0.01:10;
y = sin(pi*n);
plot(n,y)
```

დროითი ბიჯის შემცირებასთან ერთად სიგნალის გრაფიკი უფრო და უფრო გლუვი ხდება. სცადეთ სხვადასხვაა ბიჯის ზომები, მაგალითად 0.5, 0.1, 0.001.

გრაფიკის და მისი ღერძების დასათაურებისთვის გამოიყენეთ ბრძანებები:

```
title('sin(\pi n),  step size = 0.1'); xlabel('n'); ylabel('sin(pi n)')
```

3 ხშირად გამოყენებული სიგნალები

ააგეთ შემდეგი უწყვეტი-დროის ფუნქციების გრაფიკები მითითებულ შუალედებზე. ერთ გრაფიკზე ორი (ან მეტი) ქვე-გრაფიკის განსათავსებლად გამოიყენეთ ბრძანება *subplot*.

$$sinc(\pi\, t) = \begin{cases} \frac{\sin(\pi t)}{\pi t} & t \neq 0 \\ 1 & t = 0 \end{cases},$$

ინტერვალზე $-10\pi < t < 10\pi$.

$$rect(t),$$

ინტერვალზე $-2 < t < 2$.

$rect(t)$ ფუნქციის გამოსათვლელად დროითი ვექტორის აგების შემდეგ (მაგალითად: t=-4:0.1:4) გამოიყენეთ ბრძანება y= abs(t)<=0.5.

პარამეტრი a-ს სხვადასხვა მნიშვნელობებისთვის აგგეთ შემდეგი ორი დისკრეტული სიგნალის გრაფიკები. სიგნალი სხვადასხვა a-სთვის ერთ გრაფაზე დაიტანეთ hold on ბრძანების გამოყენებით.

$$a^n(u[n] - u[n-5]),$$

$$\cos\left(\frac{\pi}{4}n\right) a^n u[n].$$

ერთეულოვანი ბიჯის, $y = u[n]$, ასაგებად შეგიძლიათ გამოიყენოოთ y = (n>0), სადაც n დროითი ინდექსების ვექტორია.

4 დისკრეტული დროის სისტემის მაგალითი

დისკრეტული დროის ციფრული სისტემები ხშირად გამოიყენება ანალოგური სიგნალების დამუშავების სისტემების ნაცვლად. ციფრული ფოტო, ვიდეო და აუდიო მასალის შენახვა, დამუშავება და გადაცემა შესაბამის ანალოგურ მეთოდებზე მეტად ეფექტიანია. ასევე, ანალოგური სისტემებისგან განსხვავებით ზოგადი გამოყენების სისტემები (მაგალითად კომპიუტერი) საშუალებას გვაძლევს უამრავი სხვადასხვა ფორმატის სიგნალი ერთი მოწყობილობით დავამუშავოთ.

შემდეგ ორ უწყვეტი-დროის სისტემას ელექტრო ინჟინერიაში მრავალი გამოყენება აქვს:

დიფერენციატორი: $y(t) = \frac{d}{dt}x(t);$

ინტეგრატორი: $y(t) = \int_{-\infty}^{t} x(\tau)d\tau.$

იმის დემონსტირებისთვის თუ როგორ შეიძლება ზოგადი უწყვეტი დროის დისკრეტული დროის ვერსია შევიმუშავოთ, განვიხილავთ დიფერენციატორის და ინტეგრატორის დისკრეტული დროის იმპლემენტაციას.

დიფერენციატორი

წარმოებულის განმარტებიდან გვაქვს,

$$y(t) = \frac{d}{dt}x(t) = \lim_{\Delta t \to \infty} \frac{x(t + \Delta t) - x(t)}{\Delta t},$$

საკმარისად მცირე ზომის T-სთვის, $y(t)$-ს ყოველ nT დროის ერთეულში დისკრეტიზაცია გვაძლევს,

$$y(nT) = \frac{x(nT) - x((n-1)T)}{T},$$

რაც უფრო მცირე ზომისაა დისკრეტიზაციის პერიოდი, უწყვეტი-დროის გაწარმოების მით უფრო კარგი მიახლოებაა მისი დისკრეტული ვერსია. დისკრეტული დროის დიფერენციატორი მოცემულია, როგორც,

$$y[n] = (x[n] - x[n-1])/T.$$

ინტეგრატორი

ინტეგრალის T პერიოდით დისკრეტიზაცცია გვაძლევს,

$$\begin{aligned} y(nT) &= \int_{-\infty}^{nT} x(\tau)d\tau \\ &= \int_{-\infty}^{(n-1)T} x(\tau)d\tau + \int_{(n-1)T}^{nT} x(\tau)d\tau \\ &\approx y\big((n-1)T\big) + x(nT)T \end{aligned}$$

დიფერენციატორის მსგავსად მიახლოება მცირე დისკრეტიზაციის პერიოდის შემთხვევაში უფრო ზუსტია. დისკრეტული დროის ინტეგრატორის ფორმულაა,

$$y[n] = y[n-1] + x[n]T.$$

ააგეთ ორივე სისტემის ბლოკ სქემა.

გამოიყენეთ დისკრეტული დროის დიფერენციატორი $x(t) = \sin(\pi t)$-ს წარმოებულის რიცხვითი გამოთვლისთვის. სცადეთ ორი სხვადაასხვა დისკრეტიზაციის პერიოდი, $T = 0.1$ და $T = 0.001$. შეადარეთ მიღებული შედეგები.

5 სხვაობიანი განტოლებები

ამ ნაწილში ორი დისკრეტული დროის ფილტრის ეფექტს გამოვიკვლევთ. პირველი ფილტრის სხვაობიანი განტოლებაა

$$y[n] = H_1\{x[n]\} = x[n] - x[n-1] \tag{2}$$

მეორე ფილტრის განტოლებაა

$$y[n] = H_2\{x[n]\} = \frac{1}{2}y[n-1] + x[n] \tag{3}$$

დაწერეთ MATLAB-ის ფუნქციები ორივე ფილტრის დასანერგად.

გამოიყენეთ თქვენი დაწერილი ფუნქციები შემდეგი ხუთი სისტემის იმპულსური მახასიათებლის გამოსათვლელად და ააგეთ თითოეული იმპულსური მახასიათებლის გრაფიკი: $\boldsymbol{H_1}, \boldsymbol{H_2}, \boldsymbol{H_1(H_2)}, \boldsymbol{H_2(H_1)}, \boldsymbol{H_1 + H_2}$. ($H_1(H_2)$ სისტემების მიმდევრობით მიბმას აღნიშნავს, სადაც შემავალი სიგნალი $x[n]$ ჯერ H_2-ში შედის, H_2-ის გამომავალი კი H_1-ში.) ინიციალიზაციისთვის ფილტრიდან (ფუნქციიდან) გამომავალი ვექტორის პირველი ელემენტის მნიშვნელობად 1 მიუთითეთ.

MATLAB-ში ციკლის გამოყენებისას სასურველია ციკლამდე გამოყოთ მეხსირება ვექტორული გამომავალი სიგნალისთვის, ბრძანებით y = zeros(1,N), სადაც N ვექტორის სიგრძეა. ამ შემთხვევაში ციკლის ყოველი იტერაციისას ვექტორის ზომის ცვლილება არ ხდება, რაც გრძელი სიგნალების შემთხვევაში გამოთვლას საკმაოდ ასწრაფებს.

ხუთივე სისტემისთვის დახაზეთ სისტემის ბლოკ-სქემები დაყოვნებების, მიმატების და ნამრავლის ოპერაციების გამოყენებით, როგორც Figure 1-შია ნაჩვენები.

6 აუდიო სიგნალების დამუშავება

თქვენი შექმნილი ფილტრის ფუნქციები შეგვიძლია აუდიო სიგნალების დამუშავებისთვის გამოიყენოთ. პირველ რიგში, მატლაბში აუდიო ფაილი უნდა დავაიმპორტოთ, ამისთვის მატლაბის ბრძანებას, [x_audio,Fs] = audioread('name.wav'), ვიყენებთ.

შემდეგი ბმულიდან გადმოტვირთეთ აუდიო ფაილი Look Around.

ბრძანების, sound(x_audio,Fs), გამოყენებით აუდიო ფაილის დაკვრაა შესაძლებელი.

H_1 და H_2 ფილტრების იმპულსური მახასიათებლის გამოყენებით გაფილტრეთ აუდიო ფაილი და მოუსმინეთ მიღებულ შედეგებს. შეადარეთ მიღებული შედეგი ორიგინალ აუდიო ფაილს და ერთმანეთს. რა ტიპის ფილტრებია 1 და 2? როგორ შეგიძლიათ ამ ეფექტების ახსნა?

პრაქტიკაში, აუდიო სიგნალების დამუშავებისთვის აუდიო სიგნალის ანათვალების დისკრეტიზაციის სიხშირის ცვლილება შეიძლება იყოს საჭირო. ამისთვის ბრძანებას, resample(audio, new_Fs, old_Fs), ვიყენებთ სადაც Fs_old და Fs_new, შესაბამისად, ძველი და ახალი დისკრეტიზაციის სიხშირეებია.

ასევე, სტერეო აუდიო ფაილის შემთხვევაში აუდიო მონაცემები ორი ვექტორისგან შედგებიან, რომლებსაც დამოუკიდებლად ვამუშავებთ.

7 სხვაობიანი განტოლებიდან სიხშირულ მახასიათებლამდე

სხვაობიანი განტოლბებიდან (2) და (3) გამოიყვანეთ თითოეული სისტემის გადაცემის ფუნქცია $H(z)$ და სიხშირული მახასიათებელი $H(e^{jw})$.

იპოვეთ სიხშირული მახასიათებლის მაგნიტუდა $|H(e^{jw})|$ და ააგეთ მისი გრაფიკი. განიხილეთ როგორ აისახება სიხშირულის მახასიათებლის მაგნიტუდის ამ ფორმით აუდიო ფაილის ფილტრაციისას მიღებულ შედეგებზე.

MATLAB-ის ბრძანების, pspectrum(x), გამოყენებით შეგვიძლია ვექტორის სიხშირული სიმძლავრის სპექტრი გამოვსახოთ. გამოიყენეთ ეს ბრძანება ორიგინალი აუდიოს და მისი გაფილტრული ვერსიების სპექტრის ასაგებად. რა კავშირია გაფილტრულ სიგნალების სპექტრასა და შესაბამისი ფილტრების სიხშირული მახასიათებლის მაგნიტუდას შორის?