Podsjetimo se…

- primjeri slike
- primjeri video signala
- nizovi simbola
- vremenski diskretni signali i otipkavanje
- kvantizacija po amplitudi i vremenu
- funkcije i kompozicija funkcija
- sustavi kao funkcije
- primjer sustava s povratnom vezom
- opis sustava pomoću blok dijagrama
- složeni sustavi

1

- danas ćemo razmotriti:
  - bezmemorijske sustave
  - eksplicitne i implicitne sustave
  - spojnu listu
  - opisivanje memorijskih sustava
  - definiciju i spajanje konačnih automata

2

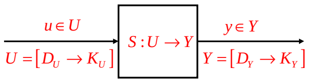
Opis sustava pomoću blok dijagrama

- u dosadašnjim primjerima već smo koristili blok dijagrame kako bi opisali sustave
- razmotrimo detaljnije slaganje više funkcijskih blokova u jedan složeniji sustav

3

Opis sustava pomoću blok dijagrama

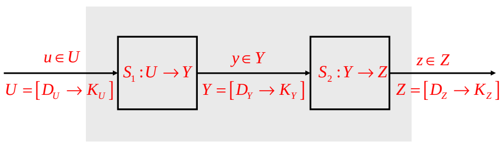
- sustav S se prikazuje blokom



$[D_U \to K_U]$
$\to [D_Y \to K_Y]$

4

Opis sustava pomoću blok dijagrama



$S : U \to Y$

- funkcija S opisuje sustav koji je nastao spajanjem sustava S_1 i S_2 u kaskadu

$$\forall u \in U, \quad S(u) = S_2(S_1(u)) \Leftrightarrow S = S_2 \circ S_1$$

5

Opis sustava pomoću blok dijagrama primjer

- primjer

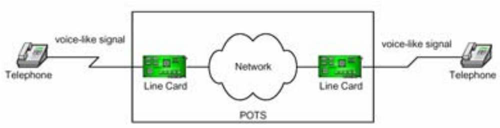
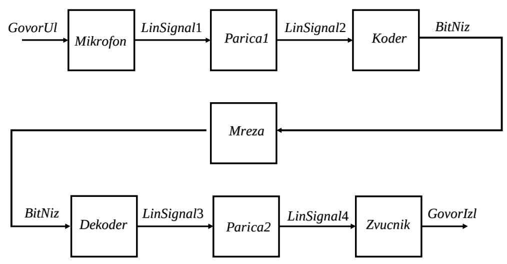


Figure 1.15: Abstraction of plain-old telephone service (POTS).

Source: Edward A. Lee and Pravin Varaiya: Structure and Interpretation of Signals and Systems, author permission

6

Opis sustava pomoću blok dijagrama primjer



7

Opis sustava pomoću blok dijagrama primjer

- neka su *Govori* skup svih mogućih ulaznih signala u telefonski mikrofon oblika

$$GovorUl : Vrijeme \to Tlak$$

- pa su *Govori* prostor signala

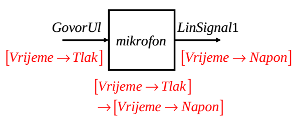
$$Govori = [Vrijeme \to Tlak]$$

- mikrofon (telefon) pretvara signal *GovorUl* u signal u skupu

$$LinSignali = [Vrijeme \to Napon]$$

8

Opis sustava pomoću blok dijagrama primjer



- pa definiramo

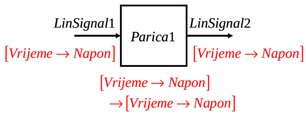
$$Mikrofon : [Vrijeme \to Tlak] \to [Vrijeme \to Napon]$$

- odnosno

$$Mikrofon : Govori \to LinSignali$$

9

Opis sustava pomoću blok dijagrama primjer



- pa definiramo

$$Parica1 : [Vrijeme \to Napon] \to [Vrijeme \to Napon]$$

- odnosno

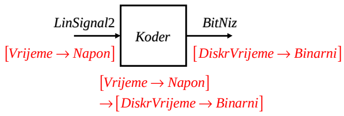
$$Parica1 : LinSignali \to LinSignali$$

10

Opis sustava pomoću blok dijagrama primjer

- pa je ulaz u *Koder*

$$(Parica1 \circ Mikrofon)(GovorUl)$$



$$Koder : [Vrijeme \to Napon] \to [DiskrVrijeme \to Binarni]$$

- odnosno

$$Koder : LinSignali \to BitNizovi$$

11

Opis sustava pomoću blok dijagrama primjer

- digitalna telefonska *Mreza* može se modelirati kao funkcija

$$Mreza : [DiskrVrijeme \to Binarni]$$
$$\to [DiskrVrijeme \to Binarni]$$

- odnosno

$$Mreza : BitNizovi \to BitNizovi$$

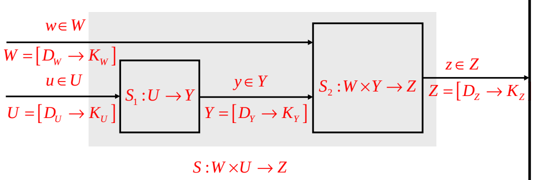
12

Opis sustava pomoću blok dijagrama primjer

- na sličan način se definiraju ostali podsustavi pa se cjelokupni put signala *GovorUl* kroz telefonsku mrežu može prikazati kao kompozicija funkcija

$$Zvucnik \circ Parica2 \circ Dekoder \circ Mreza \circ Koder \circ Parica1 \circ Mikrofon$$

Opis sustava pomoću blok dijagrama

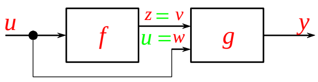


$$\forall (w,u) \in W \times U, \quad z = S(w,u) = S_2(w, S_1(u))$$

Složeni sustavi

- sustavi od kojih je sastavljen složeni sustav zovu se podsustavi
- dva i više objekata ili podsustava mogu se spojiti tako da zajedno čine jedan složeni sustav
- sustav s više ulaza i izlaza može se jednostavno razložiti na više sustava s više ulaza, ali samo s po jednim izlazom

Spajanje funkcijskih blokova u sustav

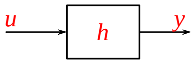


$$z = f(u)$$
$$y = g(v, w)$$

- jednadžbe spajanja:

$$\left.\begin{aligned} v &= z \\ w &= u \end{aligned}\right\} \quad y = g(f(u), u)$$

$$y = h(u)$$



jedan funkcijski blok

16

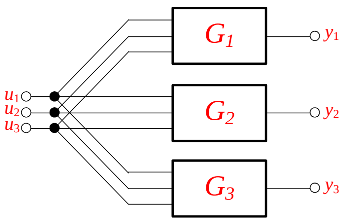
Pravila spajanja

- izlazi iz blokova se ne spajaju međusobno
- svaki ulaz bloka spaja se na izlaz nekog bloka ili je ulaz u spojeni složeni sustav
- svi ulazi podsustava su angažirani
- izlaz bloka može biti izlaz složenog sustava
- najmanje jedan izlaz podsustava je izlaz spojenog sustava

17

Složeni sustavi

- složeni sustav:



18

Spajanje sustava

- neka sustavi S_1 i S_2 imaju ulaze

  $\{u_{1i}$ za $i = 1, 2, \ldots, m_1\}$ i $\{u_{2j}$ za $j = 1, 2, \ldots, m_2\}$

  i izlaze

  $\{y_{1k}$ za $k = 1, 2, \ldots, r_1\}$ i $\{y_{2l}$ za $l = 1, 2, \ldots, r_2\}$.
- sustavi S_1 i S_2 su spojeni ako je barem jedna varijabla $u_{1i}(t)$ ulaza sustava S_1 izjednačena s jednom varijablom $y_{2k}(t)$ izlaza sustava S_2 za svaki t tj.

$$u_{1i}(t) = y_{2l}(t),\ i \in [1, m_1] \text{ i } l \in [1, r_2]$$

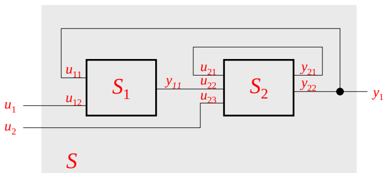
$$u_{2j}(t) = y_{1k}(t),\ j \in [1, m_2] \text{ i } k \in [1, r_1]$$

Primjer

- neka je $m_1 = 2$, $m_2 = 3$, $r_1 = 2$, $r_2 = 3$, $m = 2$ i $r = 3$.

$$y_{11} = F_{11}(u_{11}, u_{12})$$
$$y_{12} = F_{12}(u_{11}, u_{12})$$
$$y_{21} = F_{21}(u_{21}, u_{22}, u_{23})$$
$$y_{22} = F_{22}(u_{21}, u_{22}, u_{23})$$
$$y_{23} = F_{23}(u_{21}, u_{22}, u_{23})$$

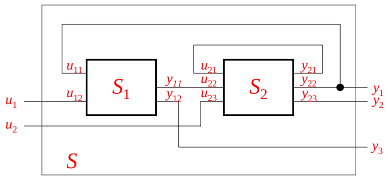
Primjer-nastavak



- jednadžbe spajanja mogu biti:

ulazi u S_1	ulazi u S_2
$u_{11}(t) = y_{22}(t)$ $u_{12}(t) = u_1(t)$	$u_{21}(t) = y_{21}(t)$ $u_{22}(t) = y_{11}(t)$ $u_{23}(t) = u_2(t)$

Primjer-nastavak



- varijable složenog sustava:

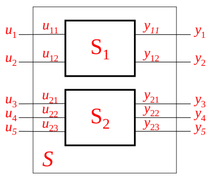
ulazi u S	izlazi S
$u_1(t)$	$y_1(t) = y_{22}(t)$
$u_2(t)$	$y_2(t) = y_{23}(t)$
	$y_3(t) = y_{12}(t)$

22

Primjer

- sustav složen od dva nezavisna podsustava:

ulazi u S	izlazi iz S
$u_1(t)$	$y_1(t) = y_{11}(t)$
$u_2(t)$	$y_2(t) = y_{12}(t)$
$u_3(t)$	$y_3(t) = y_{21}(t)$
$u_4(t)$	$y_4(t) = y_{22}(t)$
$u_5(t)$	$y_5(t) = y_{23}(t)$



- ulazi u_1, u_2 ne djeluju na izlaze y_3, y_4, y_5.
- ulazi u_3, u_4, u_5 ne utječu na izlaze y_1 i y_2.

23

Primjer-nastavak

- složeni sustav zadanih ili željenih svojstava može se dobiti sintezom jednostavnih sustava u kompleksniji.
- razlaganjem ili analizom sustava možemo dobiti dobar uvid u vladanje sustava i utjecaj pojedinih podsustava.

24

Bezmemorijski sustavi

- bezmemorijski sustav ima svojstvo da odziv sustava ovisi samo o trenutnoj vrijednosti ulaznog signala a ne o njihovim prethodnim vrijednostima
- sustav

$$F : [Realni \to Y] \to [Realni \to Y]$$

je bezmemorijski ako postoji funkcija

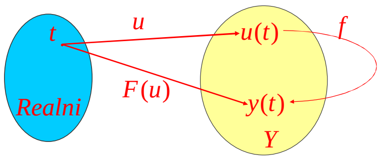
$$f : Y \to Y$$

tako da vrijedi

$$\forall t \in Realni \wedge \forall u \in [Realni \to Y], (F(u))(t) = f(u(t))$$

25

Bezmemorijski sustavi



$$y = F(u) \Rightarrow \quad y(t) = (F(u))(t) = f(u(t))$$

- bezmemorijski sustav ima svojstvo da izlaz u trenutku *t* ovisi samo o vrijednosti ulaznog signala u trenutku *t.*

26

Bezmemorijski sustavi

- razmotrimo sustav

$$Kvadrat : [Realni \to Realni] \to [Realni \to Realni]$$

gdje ako je

$$y = Kvadrat(u)$$

tada

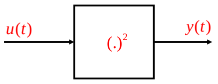
$$\forall t \in Realni, y(t) = (u(t))^2$$

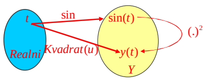
- neka je ulazni signal

$$\forall t \in Realni, u(t) = \sin(t)$$

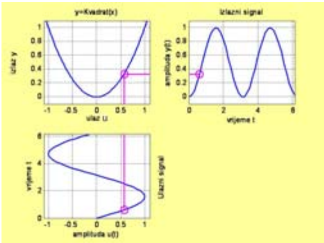
27

Bezmemorijski sustavi





$y(t) = (Kvadrat(x))(t) = (\sin(t))^2$



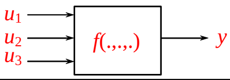
$$y(t) = (\sin(t))^2$$
$$= \frac{1}{2}(1-\cos(2t))$$

Funkcijski blok

- kod bezmemorijskih sustava izlaz u trenutku *t* ovisi samo o vrijednosti ulaznog signala u trenutku *t*.
- elementi sustava su prikazani funkcijskim blokom.
- funkcijski blok je opisan funkcijom.

$$y(t) = f(u_1(t), u_2(t), \dots, u_m(t))$$

$y(t), u_i(t) \in Realni$



Funkcijski blok s jednim ulazom i jednim izlazom

- $y(t) = f(u(t))$, za svaki t
- f je funkcija koja broju pridružuje broj.
- blok označen s f nazivamo funkcijski blok.
- funkcijska veza ulaza i izlaza može se dati:
  - analitičkim izrazom pomoću poznatih funkcija
  - krivuljom u *u-y* ravnini
  - tablicom diskretnih vrijednosti

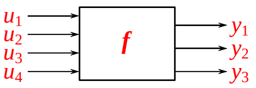
MATLAB

Primjeri funkcijskih blokova s jednim ulazom i jednim izlazom

Spajanje funkcijskih blokova u sustav

- sustav s više ulaza i više izlaza:

$y_1 = f_1(u_1,u_2,u_3,u_4)$
$y_2 = f_2(u_1,u_2,u_3,u_4)$
$y_3 = f_3(u_1,u_2,u_3,u_4)$



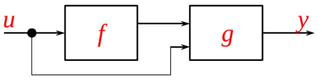
uvođenjem vektora:

ulaz: $[u_1, u_2, \dots u_m]^T$
izlaz: $[y_1, y_2, \dots y_r]^T$ } $\mathbf{y} = \boldsymbol{f}(\mathbf{u})$ ... gdje je $\boldsymbol{f}$ vektorska funkcija

Eksplicitni i implicitni sustavi

- dvije grupe sustava bez memorije:
  - eksplicitni sustavi,
  - implicitni sustavi.
- podjela prema tome da li signal na svom putu kroz sustav čini petlju:
  - eksplicitni sustav – nema petlje,
  - implicitni sustav – ima jednu ili više petlji.

Prikaz sustava listom spajanja



g: *f*, *u*
f: *u*

- svaki funkcijski blok ima jedan redak u listi.
- izlazne varijable označene su oznakom funkcijskog bloka.
- ulazne varijable označene su:
  - ulazima u dotični blok,
  - oznakama bloka čiji izlaz ulazi u dotični blok.

Prikaz sustava listom spajanja

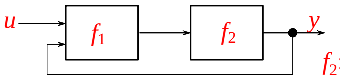
- Sortirana lista:
  - kada su redci u listi spajanja složeni tako da u svakom retku ime funkcije ili varijable desno od dvotočke možemo naći lijevo od dvotočke negdje iznad tog retka ili je to ulaz sustava, kažemo da je lista sortirana.

g: *f*, *u*
f: *u*
nesortirana lista

⇨

f: *u*
g: *f*, *u*
sortirana lista

Prikaz sustava listom spajanja



f_2: f_1
f_1: f_2, u

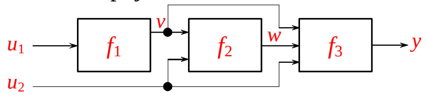
ili

f_1: f_2, u
f_2: f_1

- lista se ne može sortirati ⇒ sustav je implicitan
- implicitni sustav je sustav s povratnom vezom
- spojna lista je način ustanovljavanja da li je sustav eksplicitni ili implicitni u slučaju da to nije moguće ustanoviti vizualnom inspekcijom

Formulacije i rješenje jednadžbi sustava (eksplicitni sustav)

- Ulazno izlazne jednadžbe na osnovu sortirane spojne liste:

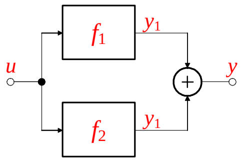


f_1: u_1
f_2: f_1, u_2
f_3: f_1, f_2, u_2

$v = f_1(u_1)$
$w = f_2(v, u_2)$
$y = f_3(v, w, u_2)$
$y = f_3\{f_1(u_1), f_2[f_1(u_1), u_2], u_2\}$

37

Formulacije i rješenje jednadžbi sustava (spajanje u paralelni slog)



$$y = y_1 + y_2$$
$$y_1 = f_1(u)$$
$$y_2 = f_2(u)$$
$$y = f(u) = f_1(u) + f_2(u)$$
$$f = f_1 + f_2$$

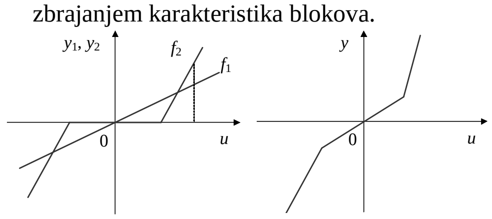
- paralelni spoj ili slog.
- veći broj sustava složenih paralelno:

$$y = f(u) = \sum_{1}^{n} f_i(u)$$

38

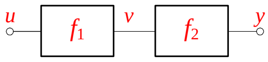
Formulacije i rješenje jednadžbi sustava (spajanje u paralelni slog)

- karakteristika paralelnog sloga dobiva se zbrajanjem karakteristika blokova.



39

Formulacije i rješenje jednadžbi sustava (spajanje u kaskadu)



$y = f_2(v)$

$v = f_1(u)$

$y = f_2(f_1(u))$

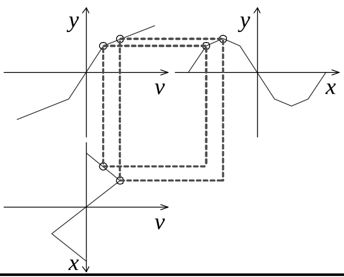
- kaskada sustava.
- funkcija kaskade je kompozicija funkcija:
  - $f = f_2 \circ f_1$
- za kaskadu s većim brojem blokova vrijedi:

$$y = f_n(f_{n-1}(f_{n-2}(\ldots(f_1(u))\ldots))).$$

$$f = f_n \circ f_{n-1} \circ f_{n-2} \circ \ldots \circ f_1.$$

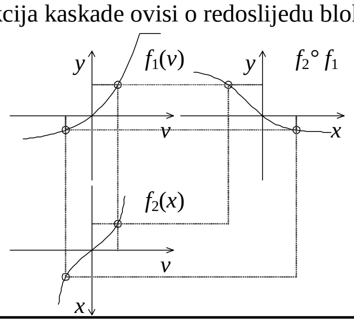
Formulacije i rješenje jednadžbi sustava (spajanje u kaskadu)

- Funkcija kaskade ovisi o redoslijedu blokova.



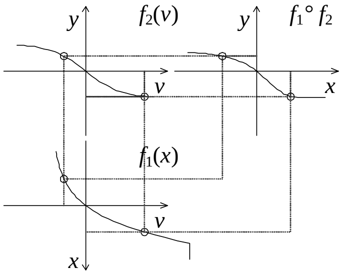
Formulacije i rješenje jednadžbi sustava (spajanje u kaskadu)

- Funkcija kaskade ovisi o redoslijedu blokova.



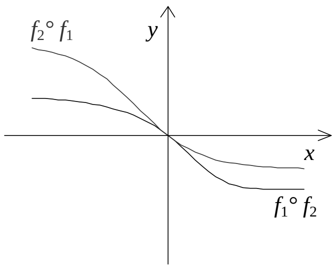
Formulacije i rješenje jednadžbi sustava (spajanje u kaskadu)

- Obrnuti redoslijed kaskada.



43

Formulacije i rješenje jednadžbi sustava (spajanje u kaskadu)



44

Bezmemorijski sustavi

- ponovimo:

  bezmemorijski sustav ima svojstvo da odziv sustava ovisi samo o trenutnoj vrijednosti ulaznog signala a ne o njihovim prethodnim vrijednostima i možemo pisati:

$$\forall t \in Realni, \quad y(t) = f(u(t))$$

45

Memorijski sustavi

- memorijski kauzalni sustav s beskonačnom memorijom definiran je s

$$\forall t \in Realni, \quad y(t) = F(u_{(-\infty,t]})$$

- pri čemu bi trivijalni slučaj $u_{(-\infty,t]} = u(t)$ učinio ovaj sustav bezmemorijskim

46

Memorijski sustavi u konačnom intervalu

- vladanje sustava uglavnom pratimo na konačnom vremenskom intervalu $(t_0, t]$ koji nazivamo interval promatranja
- zanima nas, dakle, odsječak odziva

$$y_{(t_0,t]}$$

kao posljedica odsječka pobude

$$u_{(t_0,t]}$$

47

Kontinuirrani memorijski sustavi - primjer

- neka je zadan vremenski kontinuirani sustav s ulazom u i izlazom y gdje je

$$\forall t \in Realni, \quad y(t) = \frac{1}{M} \int_{t-M}^{t} u(\tau) d\tau$$

- uz zamjenu varijabli slijedi

$$y(t) = \frac{1}{M} \int_{0}^{M} u(t-\tau) d\tau$$

- očigledno radi se o memorijskom sustav koji inače ima svojstvo izglađivanja ulaznog signala

48

Diferencijalne jednadžbe

- sustavi koji su opisani funkcijom

$KontSustavi : KontSignali \to KontSignali$

- $KontSignali$ je skup vremenski kontinuiranih signala koji ovisno o konkretnom sustavu mogu biti

$KontSignali = [Vrijeme \to Realni]$ ili

$KontSignali = [Vrijeme \to Kompleksni]$

uz $Vrijeme = Realni$ ili $Vrijeme = Realni_+$

49

Diferencijalne jednadžbe

- ovako definirana klasa sustava naziva se vremenski kontinuirani sustavi
- vrlo često se vremenski kontinuirani sustavi opisuju diferencijalnim jednadžbama

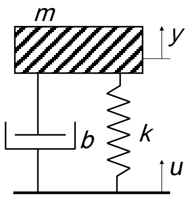
50

Diferencijalne jednadžbe

- podsjetimo se primjera

y – pomak automobila

u – je visina ceste (neravnine)



$$m \cdot y'' = b(u' - y') + k(u - y)$$

$$y'' + \frac{b}{m} y' + \frac{k}{m} y = \frac{b}{m} u' + \frac{k}{m} u$$

uz zamjenu $2\alpha = \frac{b}{m}$ $\omega_0^2 = \frac{k}{m}$

$$y'' + 2\alpha y' + \omega_0^2 y = 2\alpha u' + \omega_0^2 u$$

51

Vremenski diskretni sustavi

- klasa sustava opisanih funkcijom
  $DiskrSustavi: DiskrSignali \to DiskrSignali$
  naziva se diskretni sustavi
- dakle vremenski diskretni sustavi djeluju s klasama diskretnih signala koji mogu biti

$DiskrSignali = [Cjelobrojni \to Realni]$ ili

$DiskrSignali = [Cjelobrojni \to Kompleksni]$ ili

$DiskrSignali = [Prirodni_0 \to Realni]$ ili

$DiskrSignali = [Prirodni_0 \to Kompleksni]$

52

Jednadžbe diferencija

- vremenski diskretni sustavi često se opisuju uz pomoć jednadžbi diferencija
- razmotrimo vremenski diskretni sustav

$S : [Prirodni_0 \to Realni] \to [Prirodni_0 \to Realni]$

gdje je za

$\forall u \in [Prirodni_0 \to Realni], \quad y = S(u)$

dan s

$$\forall n \in Cjelobrojni, \quad y(n) = \frac{1}{3}(u(n) + u(n-1) + u(n-2))$$

53

Jednadžbe diferencija

- izlaz za svaki indeks n je srednja vrijednost tri uzlazne vrijednosti (tri uzorka)
- ovo je jednostavni sustav za usrednjavanje i naziva se *moving average*

54

Jednadžbe diferencija

- neka je $u(n)$ = $s(n)$, jedinični skok, funkcija definirana kao

$$\forall n \in Cjelobrojni, \quad s(n) = \begin{cases} 1 & \text{za} \quad n \geq 0 \\ 0 & \text{za ostale } n \end{cases}$$

- rješavanjem zadane jednadžbe diferencija izračunava se izlaz y

$$\forall n \in Cjelobrojni, \quad y(n) = \begin{cases} 1 & \text{za} \quad n \geq 2 \\ 2/3 & \text{za} \quad n = 1 \\ 1/3 & \text{za} \quad n = 0 \\ 0 & \text{inače} \end{cases}$$

55

Jednadžbe diferencija

- opći oblik sustava za usrednjavanje tzv. *moving average system* je

$$\forall n \in Cjelobrojni, \quad y(n) = \frac{1}{M}\sum_{j=0}^{M-1} u(n-j)$$

- za svaki n ovaj sustav daje srednju vrijednost od zadnjih M vrijednosti ulaznog signala
- ovaj sustav predstavlja diskretnu realizaciju sustava opisanog s

$$y(t) = \frac{1}{M}\int_0^M x(t-\tau)d\tau$$

56

Deklarativne i imperativne definicije sustava

- dani primjeri su bili primjeri deklarativne definicije sustava
- imperativna definicija predstavlja postupak za izračunavanje izlaznog signala za zadani ulazni signal

57

Deklarativne i imperativne definicije sustava

- imperativni opis kontinuiranih sustava moguć je uz pomoć analognih el. mreža ili ostalih fizikalnih sustava koji su kontinuirani sami po sebi
- imperativni opis sustava može biti u obliku niza naredbi odgovarajućeg programa u nekom od programskih jezika

58

Deklarativne i imperativne definicije sustava

- u slučaju kontinuiranih sustava biti će potrebne odgovarajuće aproksimacije koja se koriste pri rješavanju diferencijalnih jednadžbi
- vremenski diskretni sustavi često se definiraju kao *automati* engleski *state machines*

59

Automati

- do sada korišten model *ulaz - izlaz* koji je definirao izlazni niz za zadani ulazni niz
- uvodi se prikaz sustava temeljen na poznavanju stanja sustava i ulaznog niza i naziva se model s varijablama stanja
- ideja se temelji na činjenici da sustav u svom djelovanju prolazi kroz niz promjena stanja
- ovo je imperativni opis sustava

60

Automati

- često je korisno opisati dio sklopovlja ili dio računalnog programa kao sustav i to korištenjem modela s varijablama stanja
- ovaj pristup omogućava bolju analizu od drugih formalnih opisa
- za početak razmatramo model s varijablama stanja za automate (sustave) s konačnim (i relativno malim) brojem stanja – konačni automati

61

Uvod u automate

- konačni automati uglavnom su vezani uz nizove događaja
- stanje sustava predstavlja (sažimlje) povijest sustava
- prema tome automat će generirati znak (uzorak) izlaznog signala na temelju ulaznog znaka (uzorka) i stanja sustava

62

Automati

- preciznije
  1. *razmatra se ulazni znak u(n)*
  2. *automat generira izlazni znak uračunavajući znak u(n) i trenutno stanje x(n)*
  3. *na temelju znaka u(n) i trenutnog stanja x(n) automat izračunava novo stanje x(n+1). Ovo se naziva prijelaz stanja, ažuriranje ili engleski **update***
  4. *automat se vraća na točku 1. postupka i uzima u razmatranje znak u(n+1)*
- na ovaj način automat specificira niz izlaznih znakova za niz znakova ulaznog signala

63

Automati

- opis sustava kao funkcije uključuje tri dijela: skup ulaznih signala, skup izlaznih signala i funkciju samu,

$$F : UlazniSignali \to IzlazniSignali$$

- u slučaju automata ulazni i izlazni signali su oblika

$$NizDogadjaja : Prirodni_0 \to Znakovi$$

gdje su $Prirodni_0 = \{0,1,2,\ldots\}$ a *Znakovi* proizvoljni skup

64

Automati

- automat se definira uređenom petorkom

$$Automat = (Stanja, Ulazi, Izlazi, FunkcijaPrijelaza, pocetnoStanje)$$

značenje ovih imena je

Stanja - *prostor stanja*

Ulazi - ulazni skup znakova ili *ulazni alfabet*

Izlazi – izlazni skup znakova ili *izlazni alfabet*

$pocetnoStanje \in Stanja$ je pocetno stanje

$$FunkcijaPrijelaza : Stanja \times Ulazi \to Stanja \times Izlazi$$

65

Automati

- *Ulazi* i *Izlazi* su skupovi mogućih ulaznih odnosno izlaznih znakova
- skup *UlazniSignali* sastoji se od svih beskonačnih nizova ulaznih znakova

$$UlazniSignali = [Prirodni_0 \to Ulazi]$$

- skup *IzlazniSignali* sastoji se od svih beskonačnih nizova izlaznih znakova

$$IzlazniSignali = [Prirodni_0 \to Izlazi]$$

66

Automati

- neka je ulazni signal

$$u \in UlazniSignali$$

- pojedini znak u signalu može se označiti

$$u(n), \quad \forall n \in Prirodni_0$$

n ovdje ne predstavlja trenutak u vremenu već korak (poziciju) u nizu

- cijeli ulazni signal je niz

$$\left(u(0), u(1), u(2), ..., u(n), ...\right)$$

67

Automati

- automat opisan petorkom

$$Automat = (Stanja, Ulazi, Izlazi, FunkcijaPrijelaza, pocetnoStanje)$$

definira funkciju

$$F : UlazniSignali \to IzlazniSignali$$

dakle

$$\forall u \in UlazniSignali \quad \Rightarrow \quad y = F(u)$$

68

Automati

- ali i definira postupak za izračunavanje ove funkcije za određeni ulazni signal
- niz stanja u pojedinim koracima, *odziv stanja*, $(x(0), x(1), \ldots)$ i izlazni signal y se konstruiraju, korak po korak, kako slijedi

$$x(0) = pocetnoStanje$$

i

$$\forall n \geq 0, \quad \left(x(n+1), y(n)\right) = FunkcijaPrijelaza\left(x(n), u(n)\right)$$

- svaki gornji izračun se naziva *reakcija* ili *odziv*

69

Automati

- pogodno je funkciju *FunkcijaPrijelaza* razložiti u dvije funkcije

  funkciju narednog stanja

  $$narednoStanje : Stanja \times Ulazi \to Stanja$$

  i *izlaznu funkciju*

  $$izlaz : Stanja \times Ulazi \to Izlazi$$

  pa pišemo $x(n+1) = narednoStanje\left(x(n), u(n)\right)$

  odnosno $y(n) = izlaz\left(x(n), u(n)\right)$

70

Automati

- zaključno

$$\forall x(n) \in Stanja \wedge \forall u(n) \in Ulazi,$$

$$\left(x(n+1), y(n)\right) = FunkcijaPrijelaza\left(x(n), u(n)\right)$$

$$= \left(narednoStanje\left(x(n), u(n)\right), izlaz\left(x(n), u(n)\right)\right)$$

71