

Matjaž Jekl, Edvard Kraševc, Juš Kocijan, Borut Zupancič

Fakulteta za elektrotehniko in računalništvo

Tržaška 25, 61000 Ljubljana

†Institut Jožef Stefan

Jamova 39, 61000 Ljubljana

PREIZKUŠANJE ALGORITMOV VODENJA S SIMULACIJO V REALNEM ČASU

REAL-TIME SIMULATION CONTROL ALGORITHMS TESTING

Ključne besede: računalniška simulacija, simulacija v realnem času, vodenje procesov, procesni vmesniki.

Key words: computer simulation, real-time simulation, process control, process interface.

Povzetek

V članku je opisan primer uporabe sposobnosti simulacije v realnem času simulacijskega jezika SIMCOS. Z njim lahko preizkusimo algoritem vodenja na realnem procesu. V ta namen je potrebno digitalnemu računalniku dodati ustrezne vmesnike za povezavo s signali procesa. V regulacijske namene je potrebno vključiti predvsem A/D in D/A pretvornik. Simulacijski jezik SIMCOS smo uporabili za preizkušanje vodenja pilotne naprave. Za regulacijski algoritem smo uporabili proporcionalno integrirni regulator, ki deluje na principu "gain scheduling" metode. Enostavnost in cenenost uporabe simulacijskega jezika SIMCOS za simulacijo v realnem času omogoča uporabo le tega tako pri raziskovalnem delu kot tudi pri izvajanju pedagoškega procesa.

Abstract

The paper deals with the possibility of real-time simulation in simulation language SIMCOS which was used for the evaluation of control algorithms for real processes. For this purpose some extra e.g. A/D and D/A converters have to be added. A proportional integral controller with gain scheduling concept was used as control algorithm. Simplicity and low price as well as fair quality are advantages in use of simulation language SIMCOS in real-time simulation. It can efficiently be used for research as well as pedagogical purposes.

1. Uvod

Simulacija v realnem času predstavlja obliko simulacije, kjer je pri simulaciji dinamičnega sistema neodvisna spremenljivka sinhronizirana z realnim časom. Ta vrsta simulacije služi za vadbe operaterjev na realnih sistemih ter preizkušanje sistemov vodenja. Poleg sinhronizacije z realnim časom je za tovrstno simulacijo značilno tudi zajemanje realnih signalov in posredovanje rezultatov v obliki realnih signalov.

Možnosti simulacije v realnem času so omejene glede na zmožnosti aparaturne opreme na kateri se ta izvaja. Z razvojem osebnih računalnikov se je povečala tudi dostopnost uporabe digitalne simulacije. Za simulacijo na osebnih računalnikih uporabljamo digitalne simulacijske jezike. SIMCOS- jezik za simulacijo zveznih in diskretnih dinamičnih sistemov je bil razvit na Fakulteti za elektrotehniko in računalništvo v sodelovanju z Inštitutom Jožef Stefan v Ljubljani.

Poleg drugih simulacijskih sposobnosti [4, 5, 7], lahko z jezikom SIMCOS izvajamo tudi simulacijo v realnem času. Glede na aparaturno opremo (osebni računalnik), ki ni posebno zmogljiva, simulacijski jezik ni namenjen za posebno hitre simulacije, vendar pa omogoča čase vzorčenja dolžine nekaj 100 ms, kar je dovolj pri številnih aplikacijah. Čeprav se je zaradi aritmetičnih koprocesorjev tudi osebnim računalnikom toliko zvečala učinkovitost, da jih s pridom uporabljamo tudi za simulacijo v realnem času, pa je v simulacijskem jeziku SIMCOS za simulacijo v realnem času vgrajen algoritem, ki je časovno manj potraten. To je Eulerjeva integracijska metoda, ki predstavlja najenostavnejši integracijski postopek. Menjava med integracijskimi algoritmi, ki so vsebovani v simulacijskem jeziku, je enostavna in zato za simulacijo v realnem času ne potrebujemo kake posebne izvedbe jezika SIMCOS.

S simulacijo v realnem času v jeziku SIMCOS smo dosegli izvedbo cenene digitalne simulacije v realnem času in s tem možnost preizkušanja zunanijh naprav, npr. mikroračunalniškega regulatorja MMC-90, ki je bil razvit na Inštitutu Jožef Stefan ali pa preizkušanja algoritmov vodenja na realnem procesu oziroma pilotni napravi. S simulacijo v realnem času lahko preizkusimo poljubno število algoritmov vodenja in tako z iterativnim postopkom pridemo do optimalnega vodenja glede na zahteve. Prav tako lahko na ta način povečamo tudi učinkovitost in kreativnost v pedagoškem procesu. Študentje lahko preizkusijo in ovrednotijo vsak svoj algoritem vodenja, ki so si ga zamislili za vodenje pilotne naprave. Tak pristop k laboratorijskim vajam je uskladitev med realnostjo, kompleksnostjo, stroški in varnostjo.

2. Povezava digitalnega računalnika z okolico

Ker je pri simulaciji v realnem času osebni računalnik priključen na zunanjo materialno opremo, mora imeti sistem tudi ustrezne periferne enote: module za zajemanje in oddajanje podatkov (A/D in D/A pretvorniki, digitalni vhodno-izhodni moduli), časovni modul, po potrebi modul za sprejem prekinitev in druge enote.

Konfiguracijo osebnega računalnika dopolnimo z ustrezno aparaturno opremo za zbiranje podatkov in vodenje procesov. Obstaja več vrst aparaturne opreme, ki se nahaja v samem računalniku (vgrajena na prosta razširitvena mesta osebnega računalnika), ali pa jo postavimo izven računalnika ter jo povežemo z osebnim računalnikom preko standardnih vmesnikov (RS232, IEEE488).

Prva možnost je vsekakor cenejša, hitrost prenašanja podatkov pa je neprimerno večja kot preko vmesnikov, saj se podatkovni in krmilni registri aparaturne opreme nahajajo v območju nad osnovnim 640 K spominskim prostorom ali pa na vhodno/izhodnih pomnilniških lokacijah osebnega računalnika.

V našem primeru smo uporabili aparaturno opremo PCI 20000 proizvajalca Burr Brown [3], ki se sestoji iz nosilne plošče ter modulov na njej, pa tudi priključnih plošč za enostavnejše priključitve vodnikov. Aparaturno opremo PCI 20000 (nosilno ploščo z moduli) vstavimo na prosto razširitveno mesto v računalniku, uporablja pa pomnilniško naslavljanje registrov (naslovi nad 640 K pomnilnikom).

Za regulacijske namene potrebujemo predvsem A/D ter D/A pretvorniške module z ustrezno programsko opremo, ki jo vključimo v SIMCOS knjižnico podprogramov (ZSLIB.LIB). Programska oprema je napisana v FORTRAN-u, le na nižjem nivoju (komunikacija med registri aparaturne opreme ter FORTRAN-skimi podprogrami) smo uporabili podprograme v zbirniku.

S pomočjo preverjanja stanja sistemskih spremenljivk digitalnega simulacijskega jezika SIMCOS moramo zagotoviti, da se podprogrami za A/D ter D/A pretvorbo izvršijo ob točno določenih trenutkih, to je ob mnogokratnikih časa vzorčenja, ki ga definiramo v SIMCOS izvornem programu s spremenljivko TS.

A/D pretvorbo smo vpeljali s funkcijskim podprogramom AD, ki ga kličemo z

$$Y = AD(ADKAN, OJAC, TS),$$

kjer je vrednost funkcije vrednost A/D pretvorbe na kanalu ADKAN z ojačenjem OJAC (programsko nastavljivo ojačenje 1, 10, 100 ali 1000), TS pa je čas vzorčenja.

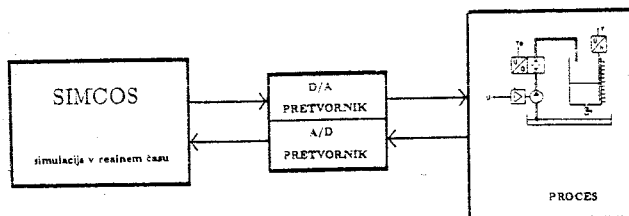
D/A pretvorba pa je regularizirana v obliki podprograma, zato jo iz SIMCOS izvornega programa kličemo iz PROCEDURAL bloka (v SIMCOS-u uporabljamo PROCEDURAL blok v

primeru klica FORTRAN-skih podprogramov). PROCEDURAL blok za klic D/A pretvorbe ima obliko

```
PROCEDURAL ( =U,DAKAN)
  CALL DA (DAKAN, U, TS)
END
```

kjer je U želena vrednost izhoda D/A kanala DAKAN, TS pa čas vzorčenja.

Na sliki 1 je predstavljena shema povezave digitalnega računalnika na katerem teče simulacija z digitalnim simulacijskim jezikom SIMCOS, s procesom.



Slika 1: Shema povezave dig. računalnika s procesom

3. Simulacija algoritma vodenja v realnem času

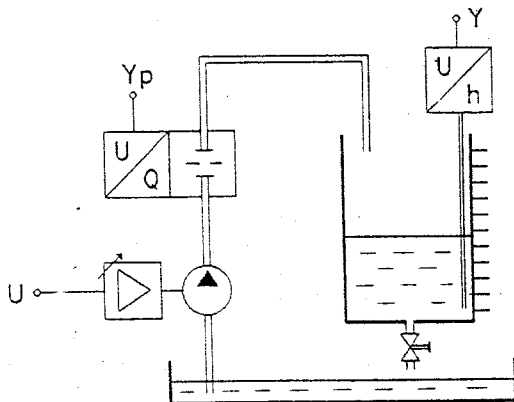
Kot primer uporabe simulacije s simulacijskim jezikom SIMCOS v realnem času, bomo predstavili realizacijo vodenja pilotne naprave [8]. Pilotno napravo sestavljata črpalka in rezervoar z izpustnim ventilom (slika 2).

Krmilna spremenljivka obravnavanega sistema je napetost na črpalki s katero spreminjamo pretok tekočine skozi črpalko in s tem vhodni pretok v rezervoar. Kot motilna veličina nastopa odprtost izstopnega ventila s katero vplivamo na velikost izstopnega pretoka tekočine iz rezervoarja. Regulirano veličino predstavlja višina nivoja tekočine v rezervoarju. Relacije med opisanimi spremenljivkami lahko predstavimo z razmeroma enostavnim matematičnim modelom [1]:

$$A \frac{dh}{dt} = q_c - k_v \sqrt{h} \quad (1)$$

$$q_c = k_c u \quad (2)$$

pri tem je:



Slika 2: Shema pilotne naprave

A - površina prečnega prereza rezervoarja,

h - višina nivoja tekočine v rezervoarju,

q_c - pretok tekočine skozi črpalko,

k_v - konstanta ventila,

k_c - konstanta črpalke,

u - napetost na črpalčki.

Linearizirani model rezervoarja v delovni točki q_{c0} in h_0 lahko predstavimo z naslednjo prenosno funkcijo:

$$G(s) = \frac{b}{s + a} = \frac{k_p}{T_p s + 1} \quad (3)$$

Pri tem sta:

$$b = \frac{1}{A}$$

$$a = \frac{k_v}{2A\sqrt{h_0}} = \frac{q_{c0}}{2Ah_0}$$

Iz enačbe lineariziranega modela rezervoarja je razvidno, da je vrednost ojačanja in časovne konstante odvisna od delovne točke, to je višine nivoja tekočine v rezervoarju. Zato si zastavimo naslednjo regulacijsko nalogo:

s PI regulatorjem želimo regulirati višino nivoja tekočine v rezervoarju tako, da bo prenosna funkcija zaprte zanke predstavljala proces drugega reda z lastno frekvenco ω_n in relativnim dušenjem ζ [2].

Če zapišemo prenosno funkcijo PI regulatorja v naslednji obliki:

$$R(s) = K_R \frac{1 + sT_i}{sT_i} \quad (4)$$

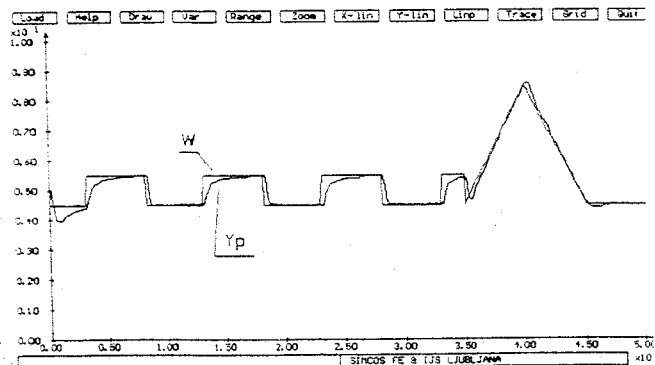
dobimo po krajšem izračunu enačbi, ki predstavljata zakone spreminjanja ojačanja in časovne konstante regulatorja, ki zagotavljata predpisano obnašanje zaključene regulacijske zanke.

$$K_R = \frac{2\zeta\omega_n - a}{b} = 2\zeta\omega_n A - \frac{q\omega_0}{2h_0} \quad (5)$$

$$T_i = \frac{2\zeta\omega_n - a}{\omega_n^2} = \frac{2\zeta}{\omega_n} - \frac{q\omega_0}{2Ah_0\omega_n^2} \quad (6)$$

Princip adaptacije parametrov glede na merljive spremembe v procesu opredeljuje literatura [2] kot "gain scheduling".

V našem primeru smo enačbi, ki opisujeta način spreminjanja parametrov PI regulatorja, realizirati na osnovi meritve pretoka $q_c = q_{c0}$ tekočine skozi črpalko. Vrednost h_0 v delovni točki pa smo nadomestili s trenutno vrednostjo referenčnega signala $h_0 = h_{ref}$. Rezultat simulacije opisanega algoritma v realnem času predstavlja slika 3. Metoda, ki smo jo uporabili pri reševanju zastavljene regulacijske naloge, predpostavlja tudi poznavanje parametrov matematičnega modela procesa. Le-te smo določili na osnovi eksperimentov na opisani pilotni napravi. Rezultat simulacije opisanega algoritma v realnem času predstavlja slika 3:



Slika 3: Zaprtozančni odziv sistema (Y_p) in referenčni signal (W)

4. Zaključek

Predstavljeni primer simulacije v realnem času s simulacijski jezikom SIMCOS ilustrira možnosti enostavne uporabe simulacijskega jezika SIMCOS za preizkušanje različnih algoritmov vodenja na realnih procesih in napravah. Enostavnost in cenenost simulacije v realnem času, ki jo nudi SIMCOS, pa samo po sebi ponuja možnost uporabe real time SIMCOS-a v pedagoškem procesu, kakor tudi pri raziskovalnem delu.

Pri tem lahko v jeziku SIMCOS simuliramo zvezni ali diskretni regulator, zvezni ali diskretni proces ter medsebojne kombinacije. Možno pa je seveda simulacija celotnega zaprtoločnega sistema na digitalnem računalniku.

Z vključitvijo FORTRAN-skih stavkov ter podprogramov, ki jih vključimo v SIMCOS knjižnico, so možnosti simulacije v povezavi z realnimi procesi ali modeli procesov skoraj neomejene (omejitev praktično predstavlja le hitrost).

Grafične možnosti, ki jih ponuja simulacijski jezik SIMCOS [6], pa omogočajo sprotno prikazovanje signalov med samim simulacijskim tekom. Pri izvedbi vodenja realnih procesov moramo upoštevati tudi določene omejitve in varnostne kriterije, ki jih je možno vključiti v simulacijski program.

Literatura

- [1] G. Stephanopoulos: Chemical Process Control, An Introduction to Theory and Practice, Prentice-Hall international series in the physical and chemical engineering sciences, Prentice-Hall, 1984.
- [2] K.J. Åström, B. Wittenmark: Adaptive Control, Addison-Wesley Publishing Company, 1989.
- [3] The Handbook of Personal Computer Instrumentation, Burr-Brown/Intelligent Instrumentation Inc., USA, 1990.
- [4] B. Zupančič, D. Matko, R. Karba, M. Šega: SIMCOS - digital simulation language with hybrid capabilities, Proceedings of the 4th symposium Simulationstechnik, Zurich, Switzerland, 205-212, 1987.
- [5] B. Zupančič, F. Bremsak, D. Matko: Simulacija zveznih in diskretnih sistemov z jezikom SIMCOS, Zbornik XXXII. jugoslovanske konference ETAN, Sarajevo, zvezek VII, 55-61, 1988.

- [6] Z. Šehić, J. Kocijan, S. Divjak, B. Zupančič, R. Karba: Primer programskega orodja za simulacijo vodenja sistemov, Zbornik XXXIV. jugoslovanske konference ETAN, Zagreb, zvezek VII, 123-130, 1990.
- [7] B. Zupančič: Discrete blocks in simulatin language SIMCOS, 2nd workshop on process automation, German - Yugoslav cooperation in scientific research and technological development, 177-183, 1987.
- [8] ELWE educational systems, experimental manual, ELWE Lehrsysteme GMBH, 1988.