

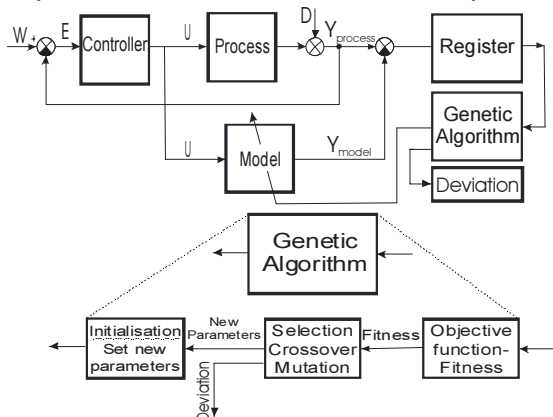
AUTODETEKCE PORUCH A POSKYTOVÁNÍ VÝUKOVÉ PODPORY NÁVRHU REGULAČNÍHO OBVODU S PROGRAMEM MATLAB-SIMULIK

Ing. David Klimánek, Doc. Ing. Bohumil Šulc, CSc.

Ústav řídicí a nástrojové techniky, FS ČVUT v Praze, Technická 4, Praha, 166 07

Abstrakt

U regulačního obvodu je detekce chyb způsobených např. změnami parametrů senzorů vždy problematická. Chyby senzorů, které nepředstavují jejich totální selhání, ale pouze odchylkou od statické charakteristiky se obtížně zjišťují, protože nejsou běžně zjevné



Obr. 1 Simulační model procesu s genetickým algoritmem

z chování regulačního obvodu. Většinou je třeba jejich počet zdvojit, příp. ztrojit, čili provést tzv. hardwarovou redundanci, což může zvýšit výrazně náklady na pořízení i na údržbu.

Ekonomičtější řešení obvykle představuje tzv. softwarová redundance, tj. vytvoření vhodného modelu na počítači, který umožní detekci vzniku změny vlastností čidla. Obvykle je třeba do tohoto modelu zajistit aktualizaci hodnot parametrů. Příspěvek popisuje, jak zmíněný problém může být řešen s použitím genetických algoritmů v prostředí Matlab –

Simulink. S detekcí poruch čidel je obvykle spojena potřeba optimálního seřízení regulačního obvodu při měnících se podmínkách. Regulační algoritmy a seřizovací postupy lze vybírat z (Šulc, Vítečková, 2004) a výhodně je zkušebně ověřovat v prostředí programu Matlab na postupně budovaném serveru tar.fs.cvut.cz.

Výhodou genetických algoritmů je nabídka možností, jak získat potřebné informace o změnách vlastností zařízení, např. čidla, na základě záznamu běžných provozních dat, které se stejně ukládají během provozu. Pro otestování těchto možností byl navržen simulační experiment používající model regulačního obvodu.

Úvod

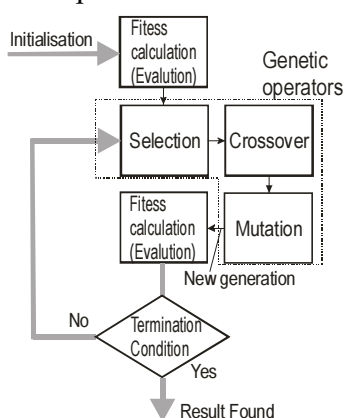
S problémem tzv. malých poruch, které sice nepředstavují totální selhání regulační funkce, ale pouze nežádoucí odchylku regulované veličiny od žádané hodnoty, se lze v praxi setkat poměrně často. Způsobují ji senzory, které nepřestávají fungovat úplně, ale pouze dodávají zkreslené hodnoty měřené veličiny. Pokud je odchylka dostatečně malá, není reálná šance, aby bez přídavných měření byla tato dysfunkce odhalena. Bezpečnou ochranu proti vzniku malých poruch poskytuje pouze zdvojení nebo ztrojení čidel, kdy např. jednoduchým algoritmem – selekce dvou ze tří hodnot lze získat informaci o chybě čidla a zajistit do doby vzniku chyby u dalšího čidla správnou hodnotu. Snaha snižování nákladů vedla na myšlenku využít modelů, které jsou schopné na základě dodávaných údajů o jiných měřených veličinách, zprostředkovat údaj o „správné“ hodnotě snímané regulované veličiny. Kvalita zprostředkování závisí na přesnosti znalosti parametrů modelu a schopnosti přizpůsobení provozním změnám. Účelem tohoto příspěvku je pokus nalézt jinou cestu k detekci zmíněných malých poruch senzorů, např. využitím genetických algoritmů.

Genetické algoritmy jako část evolučních algoritmů jsou počítačové simulační techniky založené na Darwinových principech. Evoluční algoritmy zahrnují genetické algoritmy, evoluční strategie, genetického programování a simulované žhání. Všechny zmíněné metody jsou heuristické, tzn. obsahují náhodnou složku.

Existuje mnoho kvalitních a současně volně dostupných aplikací schopných řešit problémy užitím genetických algoritmů např. aplikace Genesis a Genitor (Fleming., Purshouse, 2001), avšak žádná aplikace nenabízí takové prostředí, které by bylo zcela kompatibilní s existujícími nástroji používanými v oblasti automatizačního řízení. Nasazení genetických algoritmů v oblasti automatizačního řízení podporuje Matlab Genetic Algorithm Toolbox (Fleming 1995). Tento toolbox umožňuje aplikování genetických algoritmů na již existující model regulačního procesu.

Popis principu genetických algoritmů

Genetické algoritmy pracují s populací potenciálních řešení zadaného problému. Každý jedinec v populaci představuje jedno možné řešení. Genetický algoritmus (*obr. 2*) začíná náhodným vygenerováním počáteční populace jedinců. Dále probíhá iterační proces výpočtu kvality jedinců, selekce, křížení a mutace. Iterační proces selekce křížení a mutace pracuje dokud není splněna ukončovací podmínka.



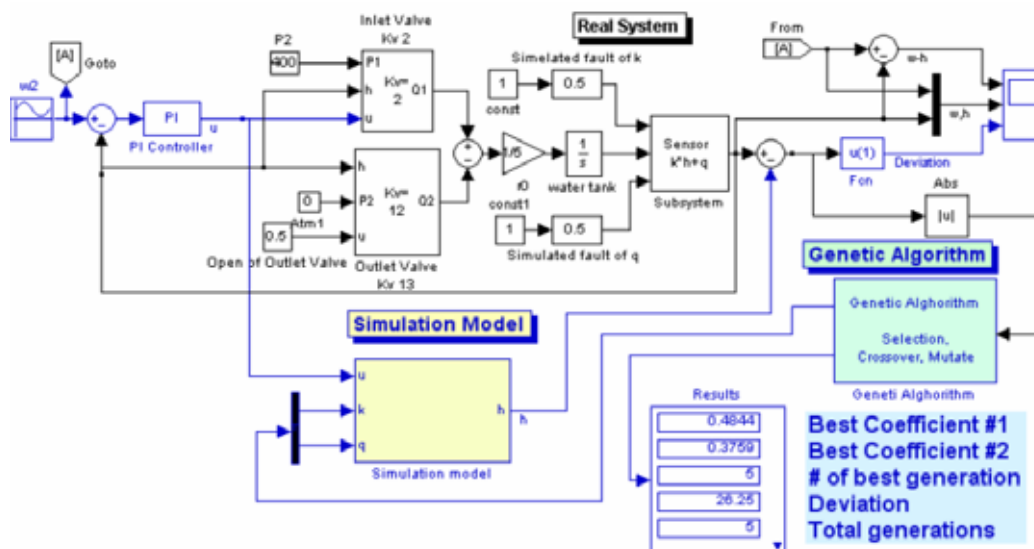
Obr. 2 Vývojový diagram genetických algoritmů

Detekce poruch s použitím genetických algoritmů

Výše popsaný genetický algoritmus byl použit v jednoduché aplikaci, kde simulovaným objektem byl model nádrže a řízenou veličinou byla výška hladiny. Vodní hladina byla řízena PI regulátorem a informace o výšce hladiny byla získávána senzorem s lineární charakteristikou. Cílem pokusu bylo ověřit, zda navržený genetický algoritmus je či není schopen detekovat simulovanou změnu parametrů regulátoru. Přestože implementace genetického algoritmu je smysluplná většinou v daleko komplexnějším případě, byl tento příklad zvolen záměrně, neboť v takto jednoduché aplikaci je ověření správné funkce algoritmu snadnější. Pro modelování chování přítokového a výtokového ventilu nádrže bylo použito tzv. motýlkové charakteristiky, která odpovídá skutečnému chování ventilu.

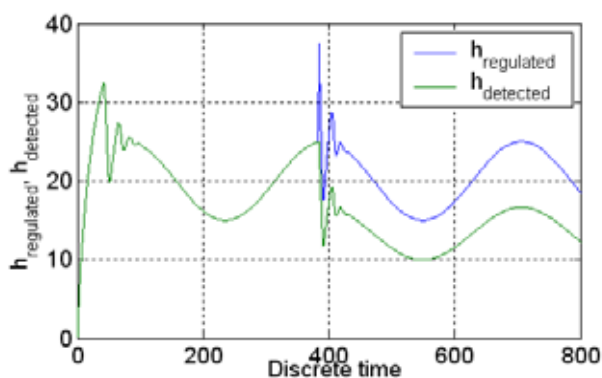
V průběhu simulace jsou porovnávány hodnoty výšek hladin modelu reálného systému a simulovaného modelu. Rozdíl těchto hodnot byl vstupním signálem bloku genetického algoritmu (*obr. 3*).

$$h_{real}(t) = kh_{system} + q \quad k, q \dots \text{parametry senzoru, } h \dots \text{výška hladiny} \quad (1)$$



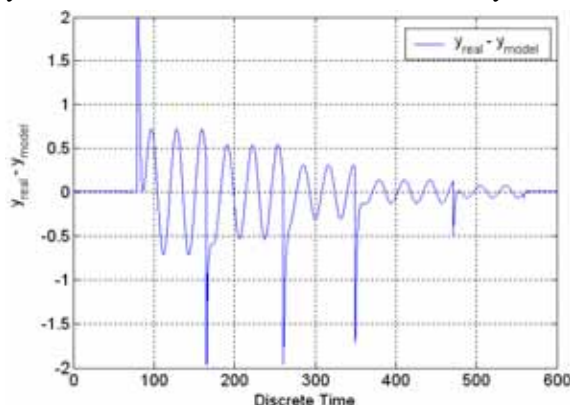
Obr. 3. Simulační schéma pro detekci poruchy

Na obr. 4 byl zobrazen vliv změny koeficientů senzoru. Nejprve senzor přenášel nezkruslenou informaci o výšce hladiny (hodnoty senzoru $k = 0,3$, $q = 0,2$), v diskrétním čase $T = 400$ nastala simulovaná změna koeficientů ($k = 0,45$, $q = 0,35$). Regulovaná a detekovaná výška hladiny se od sebe liší, senzor poskytuje zkruslenou informaci.



Obr. 4 Důsledek změny koeficientů na informaci o výšce hladiny

V průběhu simulace se genetický algoritmus snaží minimalizovat odchylku mezi reálným systémem a simulovaným modelem postupným nastavováním koeficientů simulovaného modelu a tím nalézt změněné hodnoty koeficientů čidla reálného modelu. Pokud jsou koeficienty simulačního modelu totožné s koeficienty čidla reálného modelu, genetický algoritmus nevyklovává žádnou funkci neboť rozdíl výšek hladin je nulový (obr. 5).



Obr. 5 Vliv vývoje parametrů na odchylku modelu a reálného systému

V diskrétním čase $T = 80$ byla simulována změna koeficientů senzoru reálného systému na hodnoty $k = 0,45$, $q = 0,35$. Úkolem genetického algoritmu je nalézt tyto změněné hodnoty. V tomto jednoduchém příkladě genetický algoritmus potřeboval k nalezení

simulované změny koeficientů 20 generací (tab. 1). Nově nalezené parametry byly $k= 0,45$ a $q= 0,23$. Algoritmus rychleji našel změnu koeficientu q než koeficientu k . Operátor je informován o nastalé změně parametrů čidla. V našem případě provedeme pouze manuální korekci tzn. snížíme koeficienty čidla reálného systému o vypočtenou korekci.

$$\begin{aligned} \text{correc}_k &= k_{\text{current}} - k_{\text{init}} = 0.45 - 0.30 = 0.15 \\ \text{correc}_q &= q_{\text{current}} - q_{\text{init}} = 0.23 - 0.20 = 0.03 \end{aligned} \quad (2)$$

Po provedení všech korekcí (diskrétní čas $T= 500$) rozdíl výšek hladin reálného a simulovaného modelu je opět minimální (obr. 5).

Tab. 1 Vývoj parametrů senzoru

Počet generací	Koeficient	
	k	q
10	0.300	0.200
20	0.389	0.220
30	0.431	0.230
40	0.448	0.230

Blok genetický algoritmus

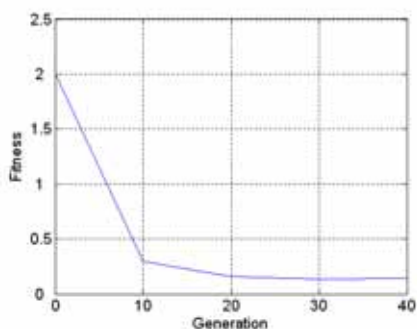
Blok genetického algoritmu pracuje s obvyklými funkcemi selekce, křížení a mutace. Pro zmíněné funkce byl použit Genetic Algorithm Toolbox (Chippefield A., Fleming, P. 1995).

Tabulka 2 představuje vývoj nejlepších koeficientů v populaci, hodnot účelové funkce a hodnotu kolikrát se daný prvek objeví v další generaci.

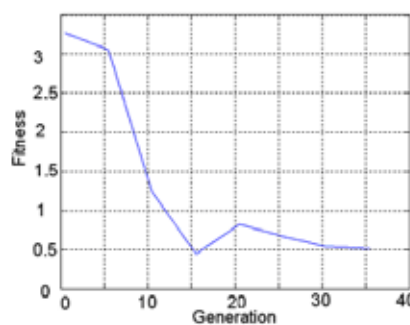
Tab. 2 Vývoj parametrů senzoru

Počet generací	Koeficient		Hodnota účelové funkce $\times 10^4$	Přepočtená účelová funkce
	k	q		
10	0.300	0.200	0.0009	2.0
20	0.389	0.220	0.8876	1.0
30	0.431	0.230	2.3823	0.5
40	0.448	0.230	0.7541	1.5

Na obr. 6 a 7 je zaznamenán vliv velikosti pravděpodobnosti mutace na velikost účelové funkce. Lepší výsledky byly dosaženy pro menší hodnotu pravděpodobnosti mutace.



Obr. 6 Průměrná hodnota účelové funkce pro 40 generací při $p_{mut}=0.1$



Obr. 7 Průměrná hodnota účelové funkce pro 40 generací při $p_{mut}=0.6$

Podpora návrhu regulačního obvodu ve výuce

Genetické algoritmy se dají s výhodou použít pro identifikaci systému, které v průběhu času mění své parametry. S přesnější identifikací systému lze dosáhnout kvalitnějšího řízení. Obecně jsou postupy seřizování a návrhů regulačního obvodu popsány v monografii (Šulc, Vítečková, 2004). Výhodou této publikace je, že mnohé z uváděných řešené příkladů mají simulační bázi v programech Matlab – Simulink a jsou postupně umísťovány na server tar.fs.cvut.cz.

Závěr

Výše popsaný algoritmus detekoval simulovanou změnu parametrů senzoru. Změněny parametrů byly nalezeny během 40 generací a doba řešení byla několik minut. Doba výpočtu závisí na počtu hledaných koeficientů a celkovém počtu senzorů, v případě složitějšího příkladu by výpočet trval mnohem déle. Detekce s použitím genetických algoritmů je zdoluhavý proces a není vhodný pro okamžitou identifikaci, kde je nezbytné dosáhnout řešení během několika iterací. V případě zmíněné aplikace tato nevýhoda nevádí, neboť tzv. malé poruchy nepředstavují totální ukončení funkce senzoru. Včasnou detekcí však snižující se kvality činnosti můžeme zamezit nepříznivým efektům, např. v oblasti spalovacích procesů nežádoucím únikům zplodin.



Další vývoj by měl být zaměřen na pokusy aplikovat chybovou detekci na aplikace, které jsou modelovány pouze lineárními bloky, a taktéž na konfrontaci výsledků dosažených jinými evolučními metodami, např. simulovaným žiháním. Výzkum bude také zaměřen na implementaci genetického algoritmu na data z reálného procesu, ověření a vnesení nových poznatků do algoritmu.

Je plánováno, že na výukový server tar.fs.cvut.cz. vedle postupně přidávaných příkladů návrhu regulačních obvodů budou zveřejněny i ukázky použití genetických algoritmů.

Použitá literatura

- CHIPPEFIELD A., FLEMING, P. 1995. *The Matlab Genetic Algorithm Toolbox*. Sheffield UK: IEE Colloquium on Applied Control Technology Using Matlab, 1995.
- FLEMING, P., PURSHOUSE, C. 2001. *IFAC Professional Brief- Genetic Algorithms in Control Systems Engineering* [online]. Sheffield UK:
<URL: http://www.oeaw.ac.at/ifac/publications/pbriefs/PB_Fleming_Purshouse.pdf >
Ann Arbor,USA: The University of Michigan Press, 1975.
- JAN, A. J. 2002. *Computing in Object Oriented Nonlinear Modeling and Control of Thermo-Fluid Dynamic Systems. Ph.D. Thesis*. Prague: Czech Technical University, 2002.
- MAŘÍK V., LAŽANSKÝ J. & ŠTĚPÁNKOVÁ O. 2001: *Umělá inteligence 3*. Praha, Academia Praha, 1993. 328 s. ISBN 80-200-0502-1.
- NEUMAN, P., ŠULC, B., ZÍTEK, P.& DLOUHÝ,T. 2000.*Non-linear Engineering Simulator of a Coal Fired Steam Boiler Applied to Fault Detection of Optimum Combustion Control*. Budapest, HU: Preprints, 4th IFAC Symposium on Fault Detection, Supervision and Safety for Technical Processes Safe process 2000, 2000.
- ŠULC, B, VÍTEČKOVÁ M, 2004. *Teorie a praxe návrhu regulačních obvodů*. Vydavatelství ČVUT v Praze, 2004, 333 s. ISBN 80-01-03007-5
- WITCZAK M. 2003. *Identification and Fault Detection of Non-Linear Dynamic Systems*. Zielona Gora, PL: University of Zielona Gora Press, 2003.
- Výukový server předmětu teorie automatizačního řízení www.tar.fs.cvut.cz

Kontaktní informace

Ústav řídicí a přístrojové techniky, Fakulta strojní ČVUT v Praze, Technická 4, Praha, 166 07
Ing. David Klimánek  david.klimanek@seznam.cz,
Doc. Ing. Bohumil Šulc, CSc.  sulc@fsid.cvut.cz