Uppgifter - doktorandkurs i diagnos VT2010

30 april 2010

Uppgift 1. I avsnitt 4 i [1] beskrivs en procedur att givet konflikter räkna ut kärndiagnoser. Baserat på beskrivningen, implementera, utan någon tanke på effektivitet, följande två funktioner:

1. en funktion som beräknar kärndiagnoser utifrån en given mängd konflikter (definierade enligt Definition 7 i [1]).
2. uppgiften ovan fast med minimala diagnoser istället för kärndiagnoser.

Väljer ni Mathematica som språk så är funktionen `LogicalExpand` mycket användbar.

Uppgift 2. Implementera både de Kleers och Reiters hitting-set algoritmer. Utvärdera komplexitet med hjälp av data som finns i filen `Tab.mat` (finns att ladda hem från hemsidan).

Filen innehåller en felkänslighetsmatrix med 301 potentiella konflikter i 15 olika komponenter. Skicka in väl valda mdelmängder av de 301 (vad är väl valda mängder?) och diskutera resultatet.

Uppgift 3. Implementera den generaliserade hitting-set algoritmen från Mattias Nybergs artikel. Utvärdera komplexitet och egenskaper med hjälp av data som finns i filen `fsm.mat` (finns att ladda hem från hemsidan).

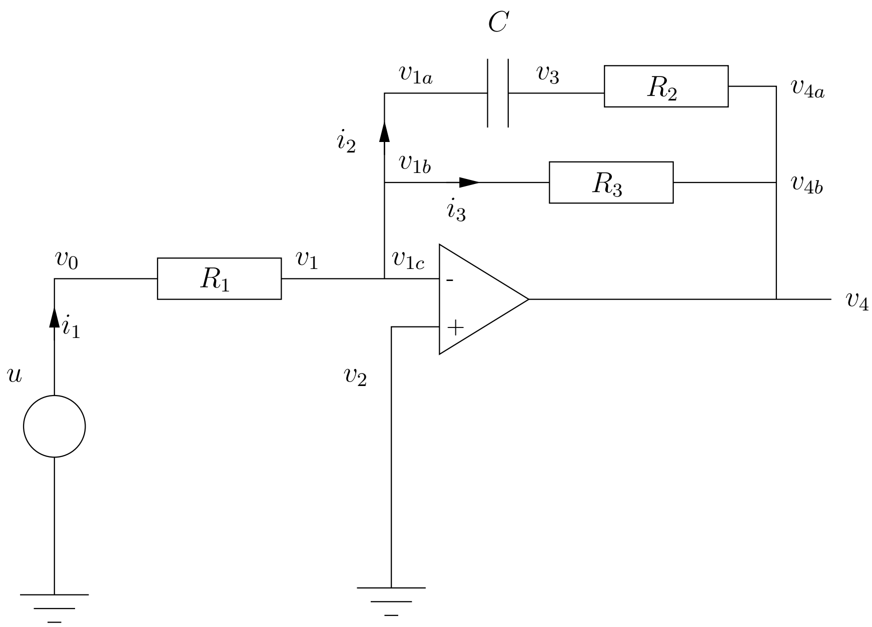
I filen så finns en variabel `fsm`, en struktur med elementen `f` samt `modestrings`. Elementet `f` är en matris där varje rad beskriver felkänsligheten hos ett diagnostest. I variabeln `modestrings` ser ni vilken beteendemod varje kolumn i `f` svarar mot. Exempelvis svarar de två första kolumnerna mot `OK[BAT2]` respektive `Degraded[BAT2]`, dvs. det felfria fallet samt ett degraderat fall hos komponenten `BAT2`.

Uppgift 4. Implementera MSO-algoritmen ifrån [5]. Det räcker att implementera grundalgoritmen (Algorithm 1), men det är såklart tillåtet att implementera den fullständiga om man känner för det.

Det rekommenderas att implementera algoritmen i Matlab, eftersom då finns Dulmage-Mendelsohn (kommando `dmperm`) färdigt att användas. Vill man implementera i annat språk så finns C-kod (samma som används i Matlab förövrigt) att hämta på http://www.cise.ufl.edu/research/sparse/CSparse/.

Testa algoritmen på data som finns i filen `SM.mat` (finns att ladda hem på hemsidan). I filen finns en strukturvariabel `SM` som har två element, `X` och `f`. Det förstnämnda är strukturen för de okända variablerna och det senare index till vilka ekvationer som påverkas av fel (10 stycken). I strukturmatrisen finns det element som är -1, dessa svarar mot variabler som är icke analytiskt lösningsbara i respektive ekvation. I den här uppgiften kan ni betrakta -1:or som 1:or.

Uppgift 5. Betrakta den enkla kretsen nedan



med komponenterna $R1$, $R2$, $R3$, C, Amp, Gen som alla kan gå sönder. En ideal modell av kretsen, med endast beskrivningar av nominellt beteende, kan skrivas med följande linjära ekvationer:

$$OK(R1) \to \ v_0 - v_1 = R_1 i_1 \qquad (e_1)$$
$$OK(R2) \to \ v_3 - v_{4a} = R_2 i_2 \qquad (e_2)$$
$$OK(R3) \to \ v_{1b} - v_{4b} = R_3 i_3 \qquad (e_3)$$
$$OK(C) \to \ C\frac{d}{dt}(v_{1a} - v_3) = i_2 \qquad (e_4)$$
$$OK(Amp) \to \ v_{1c} = v_2 \qquad (e_5)$$
$$OK(Gen) \to \ v_0 = u \qquad (e_6)$$

$$v_2 = 0 \qquad (e_7)$$
$$i_1 = i_2 + i_3 \qquad (e_8)$$
$$v_1 = v_{1a} \qquad (e_9)$$
$$v_1 = v_{1b} \qquad (e_{10})$$
$$v_1 = v_{1c} \qquad (e_{11})$$
$$v_4 = v_{4a} \qquad (e_{12})$$
$$v_4 = v_{4b} \qquad (e_{13})$$

där u är en känd styrsignal till kretsen.

a) Visa att alla felen är detekterbara om en sensor som mäter i_2 läggs till.

b) Antag att endast potentialer kan mätas, dvs. endast variabler v_k i modellen ovan. Vilka fel kan detekteras och vilken är den maximala enkelfelsisolerbarheten?

c) Gör om b-uppgiften fast nu ska även dubbelfel beaktas. Ni får även gärna utöka analysen till multipelfel av högre ordning än 2.

Uppgifterna löses med fördel baserat på resultaten i [4] och med hjälp av polynomial toolbox i Matlab. Ni som inte har tillgång till den, säg till kursledningen så fixar vi.

Uppgift 6. Verifiera uttrycket (4) i [3] för ett (godtyckligt) system via Monte-Carlo simuleringar. Välj lämpligtvis $n = 5$, $r = 2$ och godtycklig matris M med full kolumnrang, välj en brusvarians.

Jämför prestanda mellan (3) och en enkelt storlekstest på Y, dvs. larma om $Y^T Y$ är större än en given tröskel J.

Uppgift 7. I [6] löses optimeringsproblem av typen

$$\min_{v_s \in P_s} J_s = \min_{v_s \in P_s} \frac{v_s H_{d,s} H_{d,s}^T v_s^T}{v_s H_{f,s} H_{f,s}^T v_s^T} \tag{1}$$

den här uppgiften handlar om att lösa sådana.

a) Betrakta optimeringsproblemet

$$\max_v \frac{v^T L v}{v^T M v} \tag{2}$$

där L och M är symmetriska matriser och $M > 0$.

För ett generaliserat egenvärdesproblem gäller att då $L^T = L$ och $M^T = M > 0$, så har matrisknippet

$$Lz - \lambda M z = 0$$

reella egenvärden $D = \text{diag}[\lambda_1 \dots \lambda_n]$ och egenvektorer $Z = [z_1 \dots z_n]$ så att

$$LZ = MZD, \quad Z^T M Z = I$$

Matlab-kommandot `[Z,D]=eig(L,M)` löser egenvärdesproblemet.

Uppgiften är att visa att $v = z_n$, där z_n är egenvektorn kopplad till det största egenvärdet, är lösningen till optimeringsproblemet.

b) För att slippa bivillkoret på v_s i (1), som inte finns i (2), kan man observera att alla $v_s \in P_s$ kan parametriseras enligt

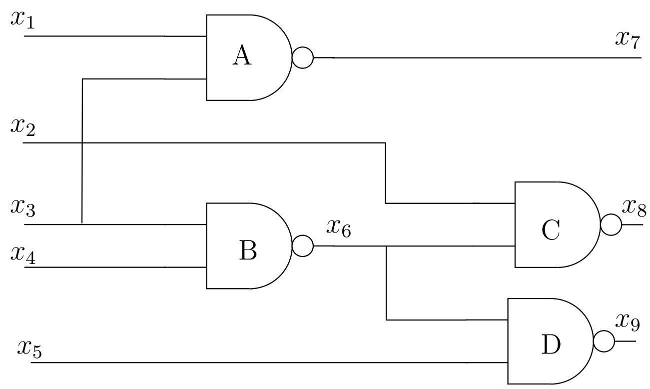
$$v_s = \zeta N$$

där N är en bas för vänster nollrum till matrisen $H_{o,s}$. Skriv om optimeringsproblemet (1) på formen (2). Matrisen N kan beräknas med Matlab-kommandot `null`.

c) Med hjälp av lösningen i a och b-uppgifterna, återskapa figur 1 och 2 i artikeln [6]. Till er hjälp finns Matlab-filen `pareq.m` att ladda hem från kurshemsidan. Kommandot skapar matriserna i ekvationerna (5) och (7) i [6] enligt

```
>> [Hu,Ho,Hd,Hf] = pareq(A, B, C, D, Bd, Fd, Ef, Ff, s );
```

Uppgift 8. Den digitala kretsen nedan ska användas för att illustrera sekvensiell diagnos enligt [2].



När felsökningen startar är $(x_1, x_2, x_3, x_4, x_5) = (1, 0, 1, 0, 1)$ given (samma i både a och b-uppgifterna)

a) Använd metodiken ifrån [2] för att hitta en mätordning. Värdena på resterande variabler är $(x_6, x_7, x_8, x_9) = (0, 0, 1, 1)$.

b) Återupprepa a-uppgiften men nu är värdena på resterande variabler $(x_6, x_7, x_8, x_9) = (1, 1, 1, 1)$.

Referenser

[1] J. de Kleer, A.K. Mackworth, and R Reiter. Characterizing diagnoses and systems. *Artificial Intelligence*, 56(2-3):197–222, 1992.

[2] Johan de Kleer and B.C. Williams. Diagnosing multiple faults. *Artificial Intelligence*, 32:97–130, 1987.

[3] M. Fouladirad and I. Nikiforov. Optimal statistical fault detection with nuisance parameters. *Automatica*, 41(7):1157–1171, 2005.

[4] Erik Frisk, Mattias Krysander, and Jan Åslund. Sensor placement for fault isolation in linear differential-algebraic systems. *Automatica*, 45(2):364–371, 2009.

[5] Mattias Krysander, Jan Åslund, and Mattias Nyberg. An efficient algorithm for finding minimal over-constrained sub-systems for model-based diagnosis. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans*, 38(1), 2008.

[6] P. Zhang, H. Ye, S.X. Ding, G.Z. Wang, and D.H. Zhou. On the relationship between parity space and h2 approaches to fault detection. *Systems & Control Letters*, 55(2):94–100, 2006.