T-79.179 **Kevät 2002**
Rinnakkaiset ja hajautetut digitaaliset järjestelmät
Laskuharjoituksen 6 vastaukset
8.3.2002

1. Muodostetaan verkon insidenssi-matriisi (transitioiden muutokset paikkojen merkintöihin pystyvektoreita):

$$C = \begin{pmatrix} -1 & 0 & 0 & 1 & 1 \\ 0 & -1 & 0 & 1 & 0 \\ 1 & -1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \end{pmatrix}$$

Jokin pystyvektori $\mathbf{f}$ on invariantti, joss $\mathbf{f}^T \cdot C = \mathbf{0}^T$, jossa $\mathbf{0}^T$ on vaakavektori.

a) Tehtävässä tulee näyttää invarianttien avulla: $M(p_4) \leq 2$.
Arvataan yhden invariantin olevan $\iota = M(p_2) + M(p_4)$. Tästä saadaan
$\mathbf{f}$-vektori, jossa molempien paikkojen kerroin on 1. Tarkistetaan että kyseinen vektori todellakin on invariantti:

$$\mathbf{f}^T \cdot C = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} -1 & 0 & 0 & 1 & 1 \\ 0 & -1 & 0 & 1 & 0 \\ 1 & -1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Siis $\iota = M(p_2) + M(p_4)$ on invariantti. Nyt lasketaan alkumerkinnästä M_0, mikä on invariantin lukuarvo kaikissa saavutettavissa merkinnöissä.

$$\mathbf{f}^T \cdot M_0 = \begin{pmatrix} 0 & 1 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} 5 \\ 2 \\ 0 \\ 0 \\ 0 \end{pmatrix} = 2$$

Siis $M(p_2) + M(p_4) = 2 \Rightarrow M(p_4) \leq 2$. Täten korkeintaan 2 prosessoria voi samanaikaisesti käyttää muistia.

b) Tehtävässä osoitetaan ensiksi $\iota = M(p_1)+M(p_3)+M(p_4)+M(p_5)$ olevan invariantti. Tarkistetaan se:

$$\mathbf{f}^T \cdot C = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} -1 & 0 & 0 & 1 & 1 \\ 0 & -1 & 0 & 1 & 0 \\ 1 & -1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & -1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Sitten lasketaan invariantin lukuarvo alkumerkinnästä:

$$\mathbf{f}^T \cdot M_0 = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 \end{pmatrix} \cdot \begin{pmatrix} 5 \\ 2 \\ 0 \\ 0 \\ 0 \end{pmatrix} = 5$$

Näin on osoitettu invariantiksi: $M(p_1)+M(p_3)+M(p_4)+M(p_5) = 5$.

2. Paikkainvarianttimenetelmä ei pysty todistamaan keskinäistä poissulkevuutta, koska algoritmi hyödyntää silmukoita. Silmukat eivät näy insidenssimatriisissa, siispä niitä ei voi kuvata invarianteilla. Kaikkien silmukoiden poistaminen tuhoaisi mutex-ominaisuuden, mutta jättäisi paikkainvariantit voimaan. Ratkaisu mutex-ongelmaan löytyy parantamalla invarianttimenetelmää *ansan* (engl. trap) käsitteellä

   Määritelmä: ansa. *Olkoon $N = (S,T;F)$ paikka-siirtymä-verkko ja olkoon $P \subseteq S$.*

   (a) P on ansa, joss $P \neq \emptyset$ ja $P^\bullet \subseteq {}^\bullet P$.

   (b) P on alustettu, joss $P \cap M_0 \neq \emptyset$.

   Määritelmä: ansaepäyhtälö. *Olkoon $N = (S,T;F)$ P/T-verkko ja olkoon $P = \{p_1,...,p_k\}$ ansa N;ssä. Tällöin*

$$p_1 + ... + p_k \geq 1$$

   on P:n epäyhtälö. P:n epäyhtälö on voimassa jos ansa on alustettu.

   Toisin sanoen: ansa on uusi Petri-verkkojen rakenteellinen ominaisuus. Joskus tällaisilla keinoilla voidaan saada tietoa verkon dynaamisista ominaisuuksista.

Tilaa testaavan mutex-algoritmin oikeellisuus

Keskinäinen poissulkevuus voidaan ilmaista epäyhtälönä

$$writing + reading \leq 1.$$

Yo. epäyhtälön todistaminen perustuu kahteen paikkainvarianttiin ja ao. ansaepäyhtälöön:

$$writing + canwrite = 1,$$

$$canread + reading = 1, \texttt{ja}$$

$$canwrite + canread \geq 1.$$

Vähentämällä ansaepäyhtälö paikkainvarianttiyhtälöistä saadaan

$$writing + canwrite + canread + reading - canwrite - canread \leq 1 + 1 - 1$$

mikä supistuu juuri siihen muotoon johon pyrimmekin.