

Petrijeve mreže v Pajku

Priprava podatkov na vhodni datoteki – podaljšek .NET

V omrežju, ki predstavlja Petrijevo mrežo, morajo biti najprej navedena vsa mesta nato pa še vsa prehajanja.

Npr:

```
*Vertices 9
```

```
1 "p1"
```

```
2 "p2"
```

```
3 "p3"
```

```
4 "p4"
```

```
5 "p5"
```

```
6 "t1"
```

```
7 "t2"
```

```
8 "t3"
```

```
9 "t4"
```

```
*Arcs
```

```
1 6
```

```
6 3
```

```
6 2
```

```
6 4
```

```
6 4
```

```
...
```

Začetni označitveni vektor podamo z razbitjem, take dimenzije kot je število mest (v prejšnjem primeru 5). Začetni označitveni vektor zgeneriramo tako, da najprej z ukazom Partition / Create Constant Partition 0 zgeneriramo prazen vektor dimenzije enake številu mest, nato pa s pomočjo File / Partition / View/Edit vnesemo željene vrednosti (število žetonov na posameznih mestih).

Pozor: čeprav govorimo o označitvenem *vektorju* je le ta v Pajku predstavljen z razbitjem (*partition*) in ne vektorjem. Razlika med razbitjem in vektorjem v Pajku je v tem, da razbitje lahko sestavljajo le pozitivna cela števila (kar je dovolj za predstavitev števila žetonov), vektorje pa sestavljajo poljubna realna števila.

Izvajanje Petrijeve mreže

Potem ko imamo prebrano omrežje in ustrezní začetni označitveni vektor, lahko poženemo izvajanje na dva načina:

1. Eno izvajanje Petrijeve mreže poženemo z

Operations / Network + Partition / Petri / Random

V primeru, da je sočasno možnih več vžigov, se vžig izbere naključno.

V posebnem oknu se nam izpišejo vsi vžigi in ustrezno zaporedje označitvenih vektorjev.

Za rezultat dobimo tudi zadnji označitveni vektor shranjen v razbitju.

2. Celotno drevo vseh možnih izvajanj Petrijeve mreže dobimo z

Operations / Network + Partition / Petri / Complete.

Globino izvajanja lahko omejimo ali pa pustimo, da se Petrijeva mreža izvaja dokler je možno (v tem primeru pustimo vrednost 0). Rezultat je hierarhija. Dvakrat kliknemo na dobljeno hierarhijo in jo pregledujemo, kot je to navada v Windowsih.

Hierarhijo spremenimo v omrežje (drevo) s

Hierarchy / Make Network. Postavljena dodatna vprašanja ne vplivajo na rezultat, zato odgovorimo kakorkoli.

Ker je drevo acikličen graf, ga lahko narišemo po nivojih, tako da najprej izračunamo nivoje z ukazom Network / Acyclic Network / Create Partition / Depth Partition / Acyclic.

Točke na istem nivoju predstavimo z istimi barvami.

Za risanje z uporabo barv uporabljamo ukaz Draw / Network + First Partition.

Če še niso, označimo točke z oznakami (Options / Mark Vertices Using / Labels).

Oznake so v tem primeru označitveni vektorji. Pravtako označimo tudi povezave z oznakami:

(Options / Lines / Mark Lines / with Labels)

Oznake povezav so v tem primeru oznake prehajanj.

Nivojsko sliko dobimo z ukazom Layers / In y Direction.

Če želimo narisati usmerjeni graf, kjer se vsak vektor pojavi le enkrat, najprej pobarvamo v dobljenem grafu vse enake vektorje z istimi barvami z uporabo ukaza Network / Create Partition / Vertex Labels.

Vse točke enake barve premaknemo naenkrat tako da z levo tipko na miški kliknemo v bližini točke in miško premaknemo (leve tipke pri tem ne spustimo).

Če ne želimo pokvariti nivojev (y koordinate) jo proglasimo za nepremično z ukazom Move / Fix / y.

Na ta način lahko preštejemo, kateri označitveni vektor se največkrat pojavi.

Vse točke enake barve stisnemo v skupno točko z ukazom Operations / Network + Partition / Shrink Network