OPIC scoring example

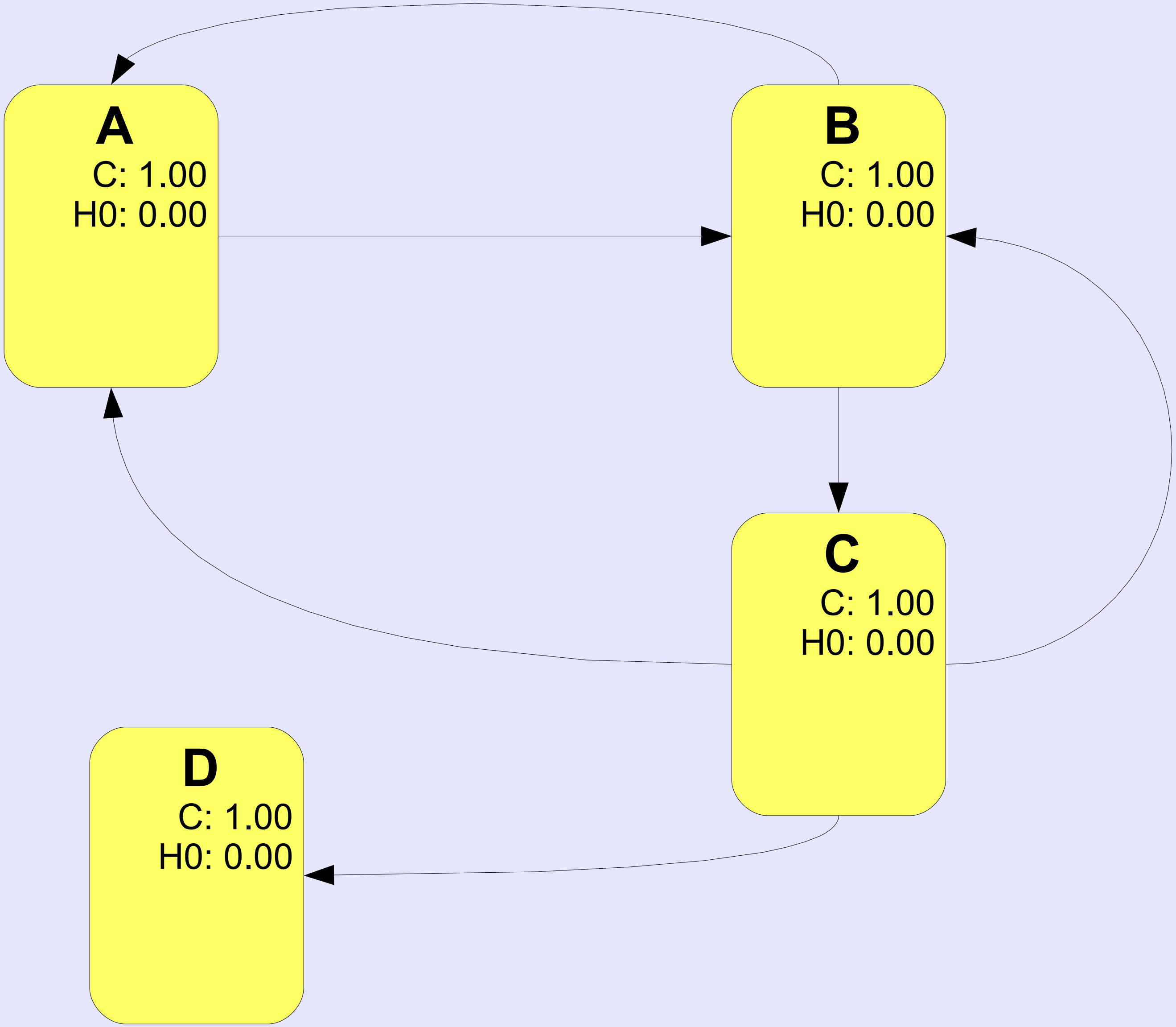
Andrzej Bialecki
<ab@getopt.org>

March 5, 2007

Scenario

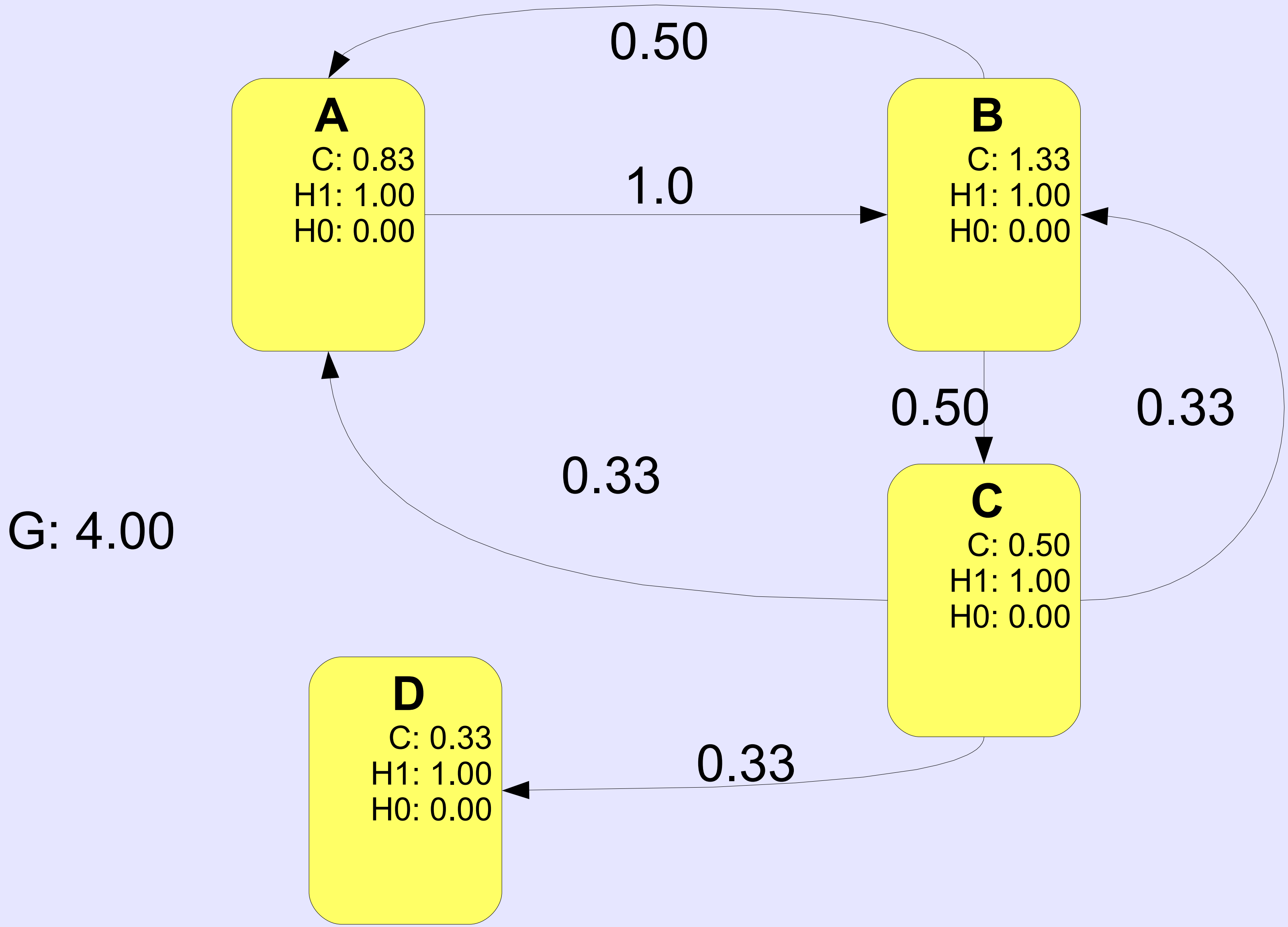
- Small test graph (immutable)
- Initial score is 1.0 for every page
- G is the total score of the distributed cash for the whole graph in each iteration
- C is the current accumulated cash per node. According to the OPIC paper this is zeroed in each iteration.
- H is the history, represents C(t-1).

OPIC, t=0

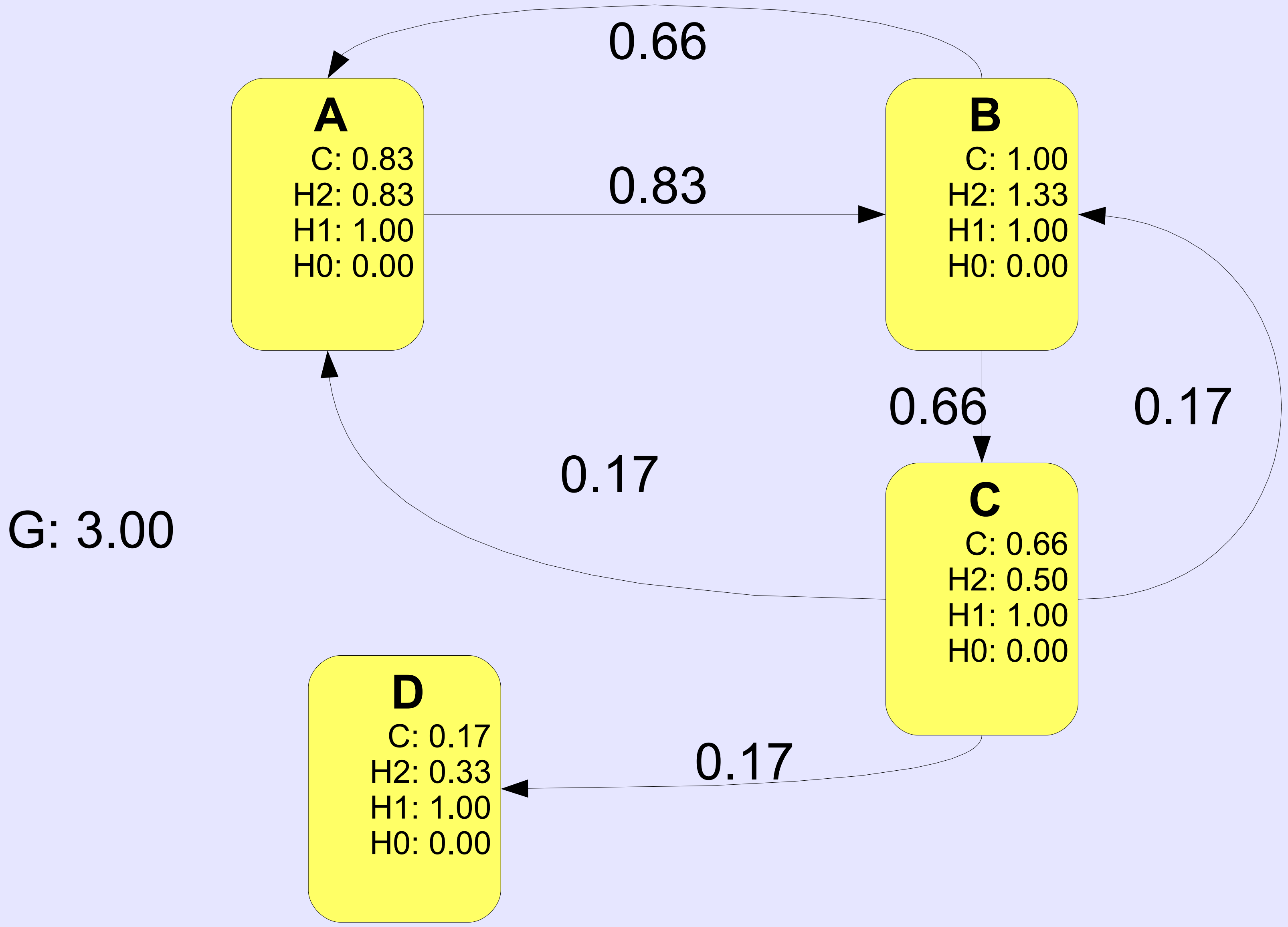


G: 0.00

OPIC, t=1

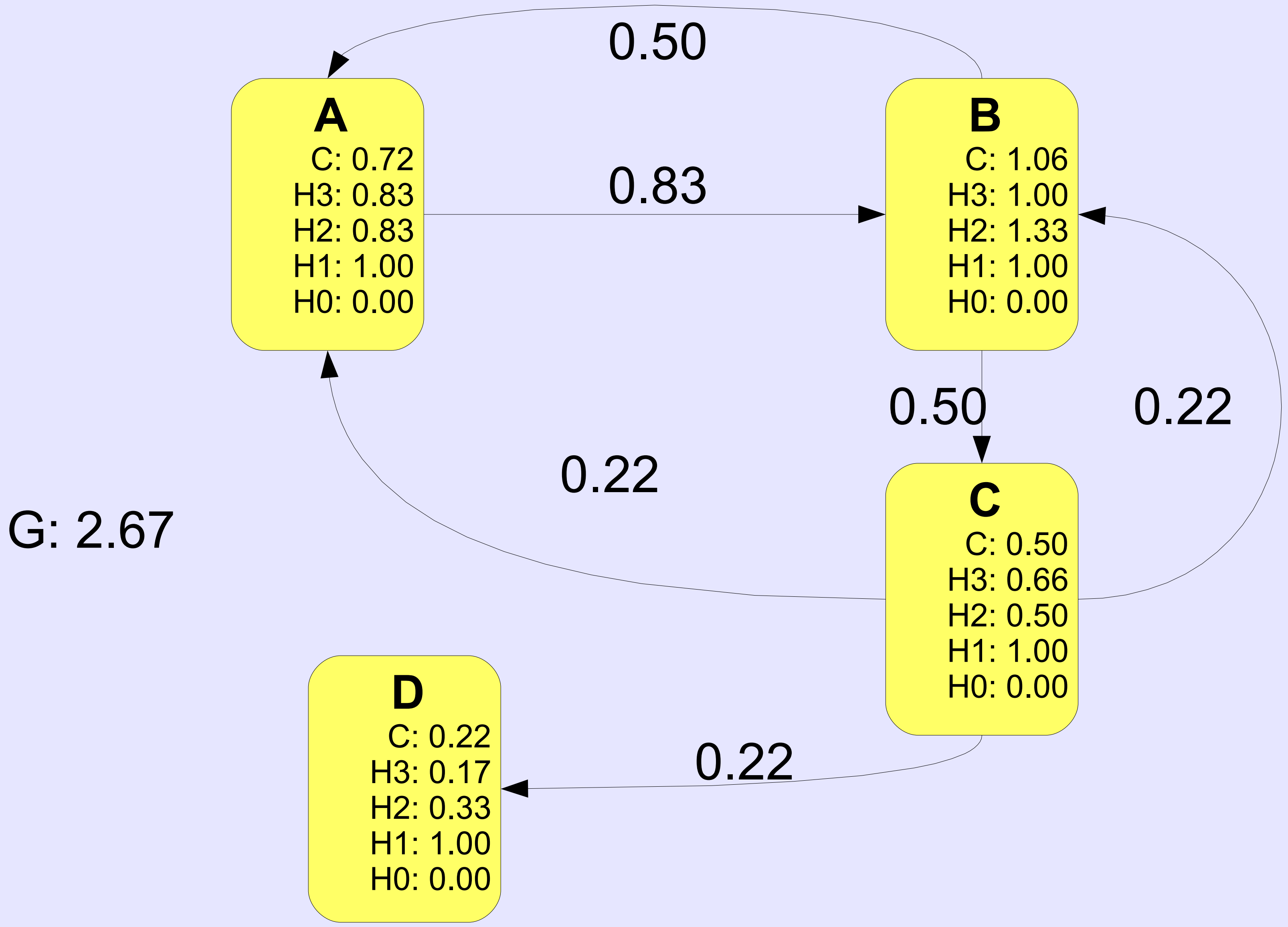

0.50
A
C: 0.83
H1: 1.00
H0: 0.00
1.0
B
C: 1.33
H1: 1.00
H0: 0.00
0.50
0.33
0.33
G: 4.00
C
C: 0.50
H1: 1.00
H0: 0.00
D
C: 0.33
H1: 1.00
H0: 0.00
0.33

OPIC, t=2

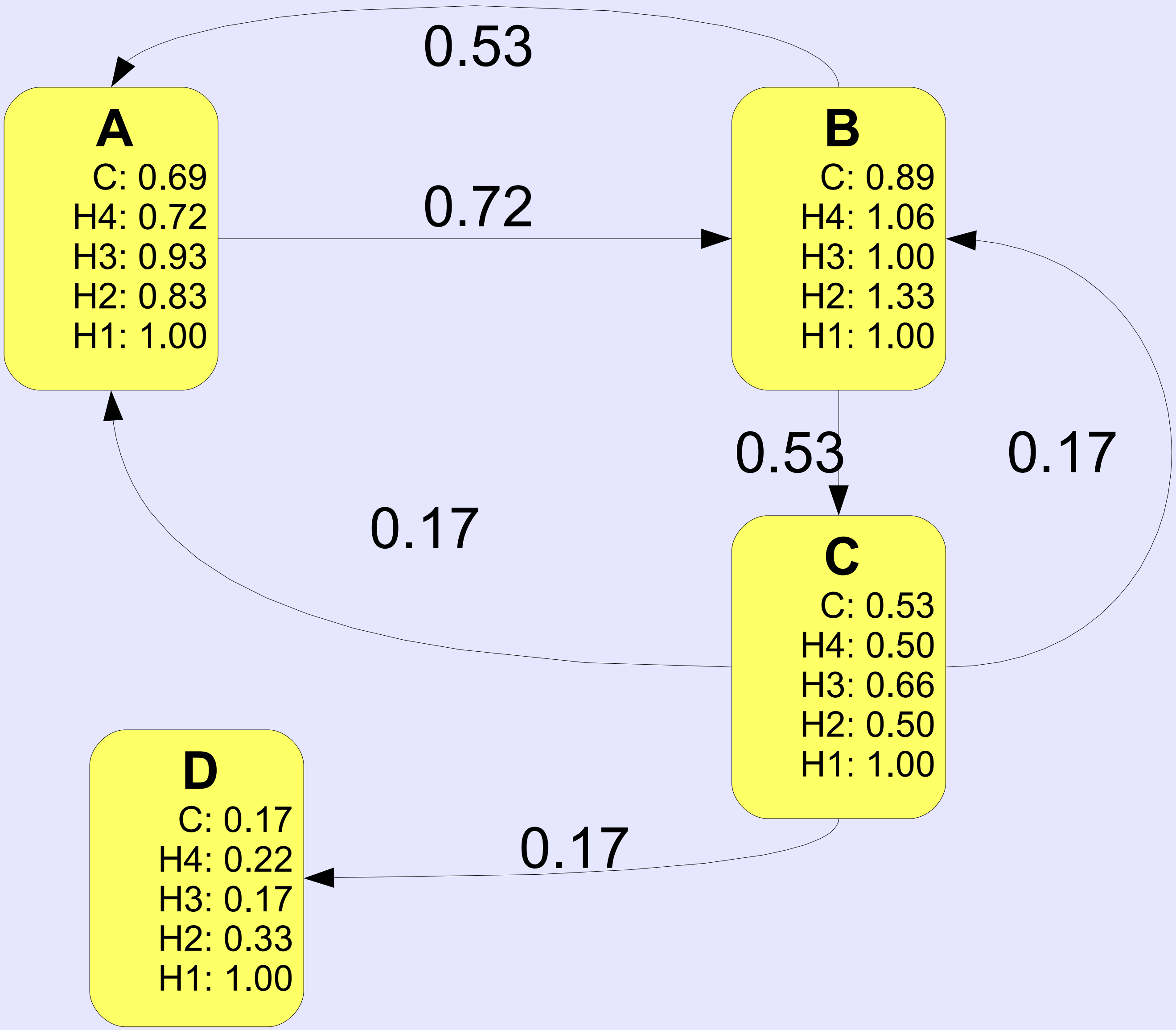


0.66
A
C: 0.83
H2: 0.83
H1: 1.00
H0: 0.00
0.83
B
C: 1.00
H2: 1.33
H1: 1.00
H0: 0.00
0.66
0.17
0.17
G: 3.00
C
C: 0.66
H2: 0.50
H1: 1.00
H0: 0.00
D
C: 0.17
H2: 0.33
H1: 1.00
H0: 0.00
0.17

OPIC, t=3

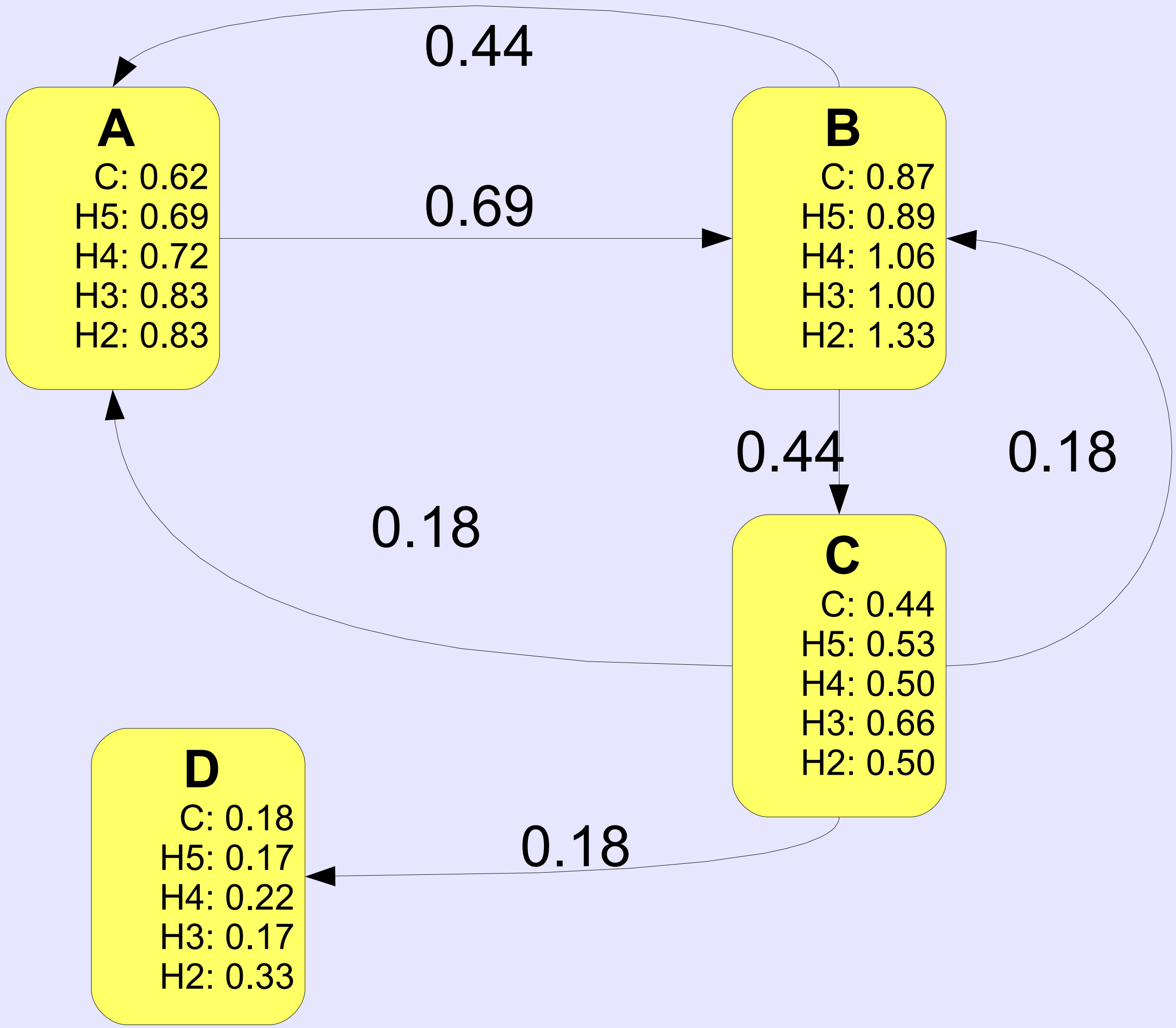

0.50
A
C: 0.72
H3: 0.83
H2: 0.83
H1: 1.00
H0: 0.00
0.83
B
C: 1.06
H3: 1.00
H2: 1.33
H1: 1.00
H0: 0.00
0.50
0.22
0.22
C
C: 0.50
H3: 0.66
H2: 0.50
H1: 1.00
H0: 0.00
G: 2.67
D
C: 0.22
H3: 0.17
H2: 0.33
H1: 1.00
H0: 0.00
0.22

OPIC, t=4


A
C: 0.69
H4: 0.72
H3: 0.93
H2: 0.83
H1: 1.00
B
C: 0.89
H4: 1.06
H3: 1.00
H2: 1.33
H1: 1.00
C
C: 0.53
H4: 0.50
H3: 0.66
H2: 0.50
H1: 1.00
D
C: 0.17
H4: 0.22
H3: 0.17
H2: 0.33
H1: 1.00
0.53
0.72
0.53
0.17
0.17
0.17

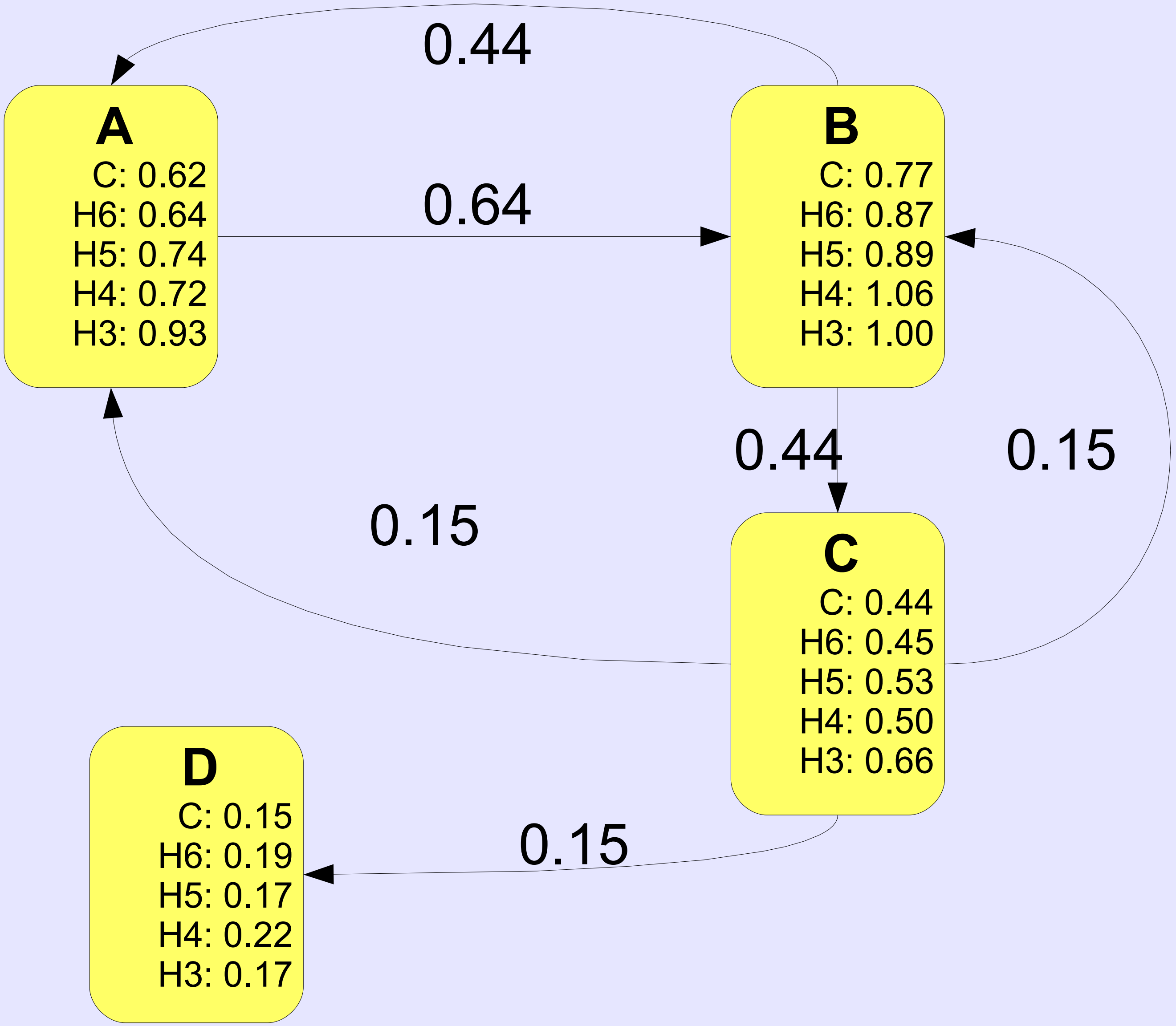
G: 2.50

OPIC, t=5


0.44
A
C: 0.62
H5: 0.69
H4: 0.72
H3: 0.83
H2: 0.83
0.69
B
C: 0.87
H5: 0.89
H4: 1.06
H3: 1.00
H2: 1.33
0.44
0.18
0.18
C
C: 0.44
H5: 0.53
H4: 0.50
H3: 0.66
H2: 0.50
D
C: 0.18
H5: 0.17
H4: 0.22
H3: 0.17
H2: 0.33
0.18

G: 2.28

OPIC, t=6



0.44

A
C: 0.62
H6: 0.64
H5: 0.74
H4: 0.72
H3: 0.93

0.64

B
C: 0.77
H6: 0.87
H5: 0.89
H4: 1.06
H3: 1.00

0.44

0.15

0.15

C
C: 0.44
H6: 0.45
H5: 0.53
H4: 0.50
H3: 0.66

D
C: 0.15
H6: 0.19
H5: 0.17
H4: 0.22
H3: 0.17

0.15

G: 2.11

Problems

Time	A [C]	A [H]	B [C]	B [H]	C [C]	C [H]	D [C]	D [H]	G
0	1.00	0.00	1.00	0.00	1.00	0.00	1.00	0.00	0.00
1	0.83	1.00	1.33	1.00	0.50	1.00	0.33	1.00	4.00
2	0.83	0.83	1.00	1.33	0.67	0.50	0.17	0.33	3.00
3	0.72	0.83	1.06	1.00	0.50	0.67	0.22	0.17	2.67
4	0.69	0.72	0.89	1.06	0.53	0.50	0.17	0.22	2.50
5	0.62	0.69	0.87	0.89	0.44	0.53	0.18	0.17	2.28
6	0.58	0.62	0.77	0.87	0.44	0.44	0.15	0.18	2.11
7	0.53	0.58	0.73	0.77	0.38	0.44	0.15	0.15	1.94
8	0.49	0.53	0.66	0.73	0.36	0.38	0.13	0.15	1.79
9	0.45	0.49	0.61	0.66	0.33	0.36	0.12	0.13	1.64
10	0.42	0.45	0.56	0.61	0.31	0.33	0.11	0.12	1.51
11	0.38	0.42	0.52	0.56	0.28	0.31	0.10	0.11	1.39
12	0.35	0.38	0.48	0.52	0.26	0.28	0.09	0.10	1.28
13	0.32	0.35	0.44	0.48	0.24	0.26	0.09	0.09	1.18
14	0.30	0.32	0.40	0.44	0.22	0.24	0.08	0.09	1.09
15	0.27	0.30	0.37	0.40	0.20	0.22	0.07	0.08	1.00

- Losing “cash”
  - D is a so called “dangling” node: black hole where our cash disappears

Fixed OPIC

Time	A [C]	A [H]	B [C]	B [H]	C [C]	C [H]	D [C]	D [H]	G
0	1.00	0.00	1.00	0.00	1.00	0.00	1.00	0.00	0.00
1	1.08	1.00	1.58	1.00	0.75	1.00	0.58	1.00	4.00
2	1.19	1.08	1.48	1.58	0.94	0.75	0.40	0.58	4.00
3	1.15	1.19	1.60	1.48	0.84	0.94	0.41	0.40	4.00
4	1.18	1.15	1.53	1.60	0.90	0.84	0.38	0.41	4.00
5	1.16	1.18	1.58	1.53	0.86	0.90	0.40	0.38	4.00
6	1.18	1.16	1.55	1.58	0.89	0.86	0.39	0.40	4.00
7	1.17	1.18	1.57	1.55	0.87	0.89	0.39	0.39	4.00
8	1.17	1.17	1.56	1.57	0.88	0.87	0.39	0.39	4.00
9	1.17	1.17	1.56	1.56	0.88	0.88	0.39	0.39	4.00
10	1.17	1.17	1.56	1.56	0.88	0.88	0.39	0.39	4.00
11	1.17	1.17	1.56	1.56	0.88	0.88	0.39	0.39	4.00
12	1.17	1.17	1.56	1.56	0.88	0.88	0.39	0.39	4.00
13	1.17	1.17	1.56	1.56	0.88	0.88	0.39	0.39	4.00
14	1.17	1.17	1.56	1.56	0.88	0.88	0.39	0.39	4.00
15	1.17	1.17	1.56	1.56	0.88	0.88	0.39	0.39	4.00

- Cash from dangling nodes spread out evenly to all other nodes (except dangling)
- Individual scores converge to stable values

Nutch implementation fixes

- Use Current and History values in a proper way
  - Record Current value in History
  - Clear current value prior to update (!)
- Collect all cash from dangling nodes
  - in ParseOutputFormat, one total per segment
- During 'updatedb' spread this cash to all other pages in the CrawlDb
- Injected and new pages should get G/I amount of cash, where I is the total number of pages
- Newly discovered pages should also get the usual cash amount from the parent node
  - Since they are initially dangling, this cash will return to all other nodes

Notes

- Abiteboul et al. suggest averaging the History either over a fixed time period, or over the last N entries (plus the Current value):

$$\boldsymbol{OPIC}[i]_t = \frac{\left(\sum_{x=t-N}^{x=t} H[i]_x\right) + C[i]_t}{(G_t + 1)}$$

- N > 0 helps to stabilize the graph
  - Accidental changes don't destroy the current score
- But N > 4 was reported to slow down the convergence