

Mapping
scientific communication
from researcher pathways

Martin Rosvall

Past

Maps of networks

Carl Bergstrom, Daniel Edler, ...

Now

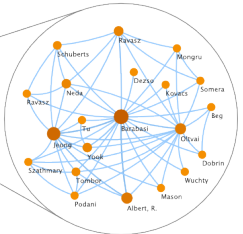
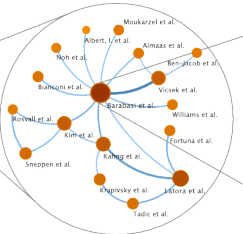
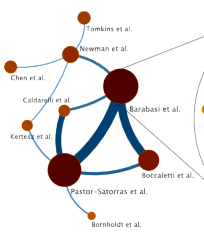
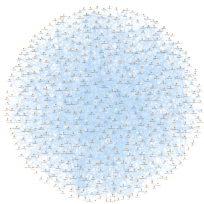
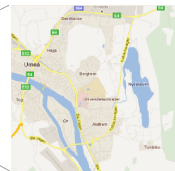
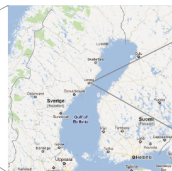
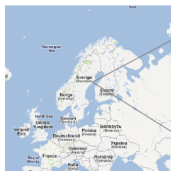
Memory networks

Renaud Lambiotte, Andrea Lancichinetti, ...

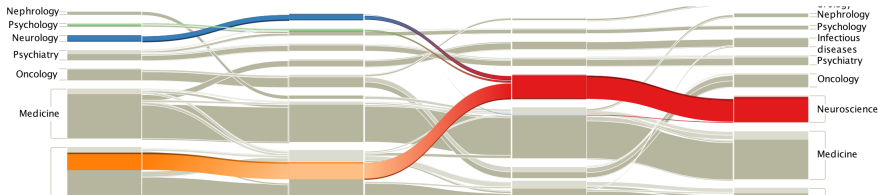
Future

Search and navigation

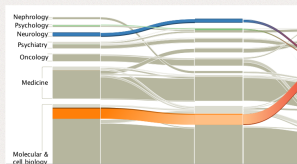
You?



Reveal structural change in complex networks



Apps »



Code »

```
using infomath::plogg;
for (unsigned int i = 0; i < numNodes; ++i)
{
  enter_log_enter += plogg(m_moduleFlowData[i].enterFlow);
  exit_log_exit += plogg(m_moduleFlowData[i].exitFlow);
  flow_log_flow += plogg(m_moduleFlowData[i].exitFlow);
  enterFlow += m_moduleFlowData[i].enterFlow;
}
enterFlow += exitNetworkFlow;
enterFlow_log_enterFlow = plogg(enterFlow);
```

Publications »

Maps of information flow reveal community structure in complex networks

Martin Rosvall and Carl T. Bergstrom
 PNAS **105**, 11118 (2008). [arXiv:0707.0609]



To comprehend the multipartite organization of large-scale biological and social systems, we introduce a new information-theoretic approach to reveal community structure in

Past

Maps of networks

Carl Bergstrom, Daniel Edler, ...

Now

Memory networks

Renaud Lambiotte, Andrea Lancichinetti, ...

Future

Search and navigation

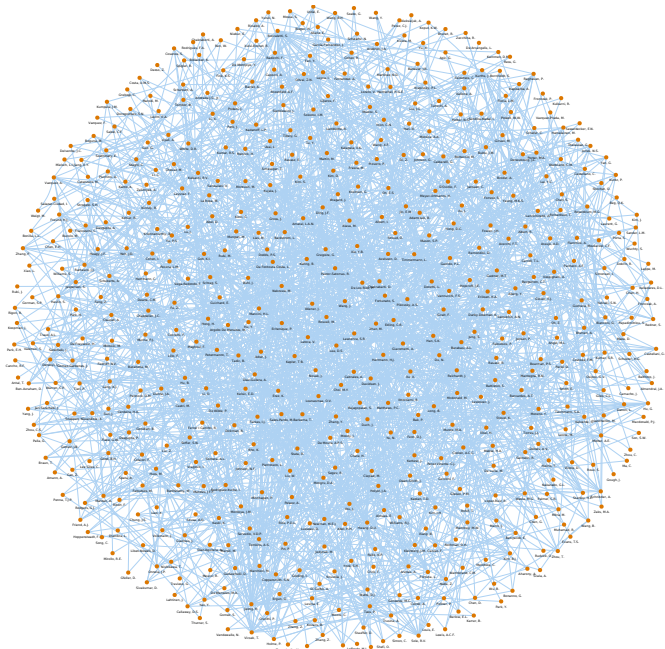
You?

How are systems organized?

How are systems organized
with respect to the flow?

Networks

Networks
describe flow
beyond nearest neighbors

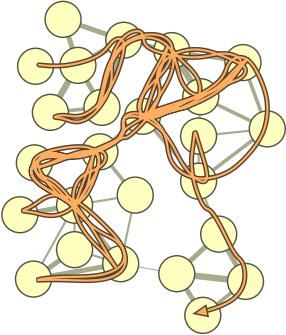


Maps

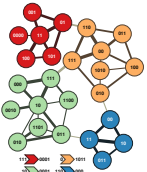
Maps

depict regularities
in the dynamics on networks
using less information

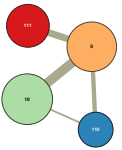
Maps by compressing flow of information on networks



1111000 1100 0110 1101 10000 11011 0110 0011 10111 1001
 0011 1001 1100 0111 10001 1110 0111 10001 0111 1110 0000
 1110 10000 0111 1110 0111 0 110 11110 1110 0000 10100 0000
 1110 10001 0111 1100 1010 1010 1100 10111 100 10000 1001 10111
 1001 0110 1001 0110 0011 0110 0011 0110 1011 0110 0011 0110
 1001 10111 0011 0100 0111 10001 1110 10000 0111 0110 10110
 111111 10110 10101 11110 00011



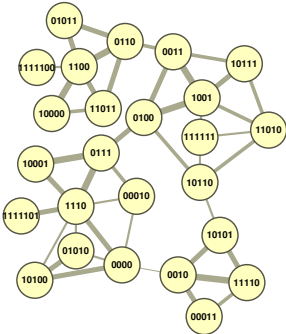
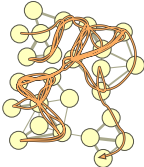
111 0000 11 01 101 100 101 01 0001 0 110 011 00 110 00 111
 1011 10 111 000 10 111 000 111 10 011 10 000 111 10 111 10
 0010 10 011 010 011 10 000 111 0000 1 111 010 000 011 00 111
 00 011 00 111 00 111 110 111 110 0011 111 010 101 00 0001 1 110
 111 00 011 110 111 101 101 10 111 000 10 000 111 0001 0 111 010
 1010 010 001 110 00 10 1011



111 000 11 01 101 100 101 01 0001 0 110 011 00 110 00 111
 1011 10 111 000 10 111 000 111 10 011 10 000 111 10 111 10
 0010 10 011 010 011 10 000 111 0000 1 111 010 000 011 00 111
 00 011 00 111 00 111 110 111 110 0011 111 010 101 00 0001 1 110
 111 00 011 110 111 101 101 10 111 000 10 000 111 0001 0 111 010
 1010 010 001 110 00 10 1011

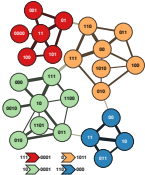
Describing flow is a coding game

– objects deserve unique names



```

1111100 1100 0110 11011 10000 11011 0110 0011 10111 1001
0011 1001 0100 0111 10001 1110 0111 10001 0111 1110 0000
1110 10001 0111 1110 0111 1110 1111101 1110 0000 10100 0000
1110 10001 0111 0100 10110 11010 10111 1001 0100 1001 10111
1001 0100 1001 0100 0011 0100 0011 0110 11011 0110 0011 0100
1001 10111 0011 0100 0111 10001 1110 10001 0111 0100 10110
111111 10110 10101 11110 00011
    
```



```

111 0000 11 01 101 100 101 01 0001 0 110 011 00 110 00 111
1011 10 111 000 10 111 000 111 10 0101 10 000 111 10 111 10
0010 10 011 010 011 10 000 111 0000 0 111 010 100 011 00 111
00 010 00 111 000 111 110 111 110 0001 110 00 100 000 0 110
111 00 011 110 101 1011 10 111 000 10 000 111 0001 0 111 010
1010 010 1011 110 00 10 011
    
```

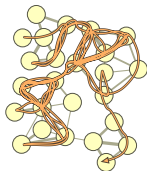


```

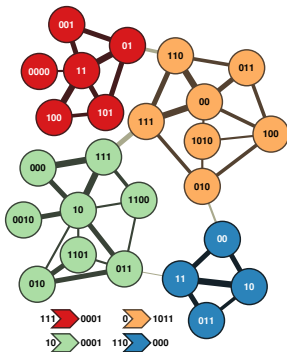
111 0000 11 01 101 100 101 01 0001 0 110 011 00 110 00 111
1011 10 111 000 10 111 000 111 10 0101 10 000 111 10 111 10
0010 10 011 010 011 10 000 111 0000 0 111 010 100 011 00 111
00 010 00 111 000 111 110 111 110 0001 110 00 100 000 0 110
111 00 011 110 101 1011 10 111 000 10 000 111 0001 0 111 010
1010 010 1011 110 00 10 011
    
```

Simplify and highlight

– important objects deserve unique names



```
111100 1100 0110 11011 10000 11011 0110 0011 10111 1001  
0011 1001 0100 0111 10001 1110 0111 10001 0111 1110 0000  
1110 10001 0111 1110 0111 1110 11101 1110 0000 10000 0000  
1110 10001 0111 0100 1010 1010 1010 1011 1001 0100 1001 1011  
1001 0100 1001 0100 0011 0100 0011 0110 1011 0110 0011 0100  
1001 1011 0110 0100 0111 10001 1110 10001 0111 0100 10110  
11111 10110 10001 11110 00011
```



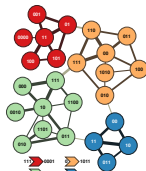
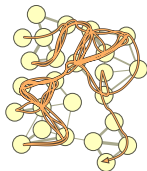
```
111 0000 11 01 101 100 101 01 0001 0 110 011 00 110 00 111  
1011 10 111 000 10 111 000 111 10 011 10 000 111 10 111 10  
0010 10 011 010 011 10 000 111 0001 0 111 010 100 011 00 111  
00 011 00 111 00 111 110 111 110 1011 111 01 101 01 0001 0 110  
111 00 011 110 111 1011 10 111 000 10 000 111 0001 0 111 010  
1010 010 1011 110 00 10 011
```



111 0000 11 01 100 000 001 00 0000 110 011 00 110 00 111
0010 10 011 010 011 10 000 111 0001 0 111 010 100 011 00 111
00 011 00 111 00 111 110 111 110 1011 111 01 101 01 0001 0 110
111 00 011 110 111 1011 10 111 000 10 000 111 0001 0 111 010
1010 010 1011 110 00 10 011

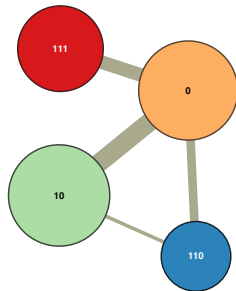
Simplify and highlight

– highlight important objects and filter away details



```
111100 1100 0110 11011 10000 11011 0110 0011 10111 1001  
0011 1001 0100 0111 10001 1110 0111 10001 0111 1110 0000  
1110 10001 0111 1110 0111 1110 11101 1110 0000 10100 0000  
1110 10001 0111 1000 1010 1010 1010 1011 1001 1001 1001 1011  
1001 0100 1001 0100 0011 0100 0011 0110 11011 0110 0011 0100  
1001 10111 0011 0100 0111 10001 1110 1000 0111 0100 10110  
11111 10110 1010 1110 00011
```

```
111 0000 11 01 101 100 101 01 0001 0 110 011 00 110 00 111  
1011 10 111 000 10 111 000 111 10 011 10 000 111 10 111 10  
0010 10 011 010 011 10 000 111 0001 0 111 010 100 011 00 111  
00 011 00 111 00 111 110 111 100 111 011 00 101 01 0000 0 110  
111 00 011 110 111 1011 10 111 000 10 000 111 0001 0 111 010  
1010 010 1011 110 00 10 011
```



```
111 0000 11 01 101 100 101 01 0001 0 110 011 00 110 00 111  
1011 10 111 000 10 111 000 111 10 011 10 000 111 10 111 10  
0010 10 011 010 011 10 000 111 0001 0 111 010 100 011 00 111  
00 011 00 111 00 111 110 111 110 1011 111 01 101 01 0001 0 110  
111 00 011 110 111 1011 10 111 000 10 000 111 0001 0 111 010  
1010 010 1011 110 00 10 011
```


The map equation

$$L(\mathbf{M}) = q_{\curvearrowright} H(\mathcal{Q}) + \sum_{i=1}^m p_{\circlearrowleft}^i H(\mathcal{P}^i)$$

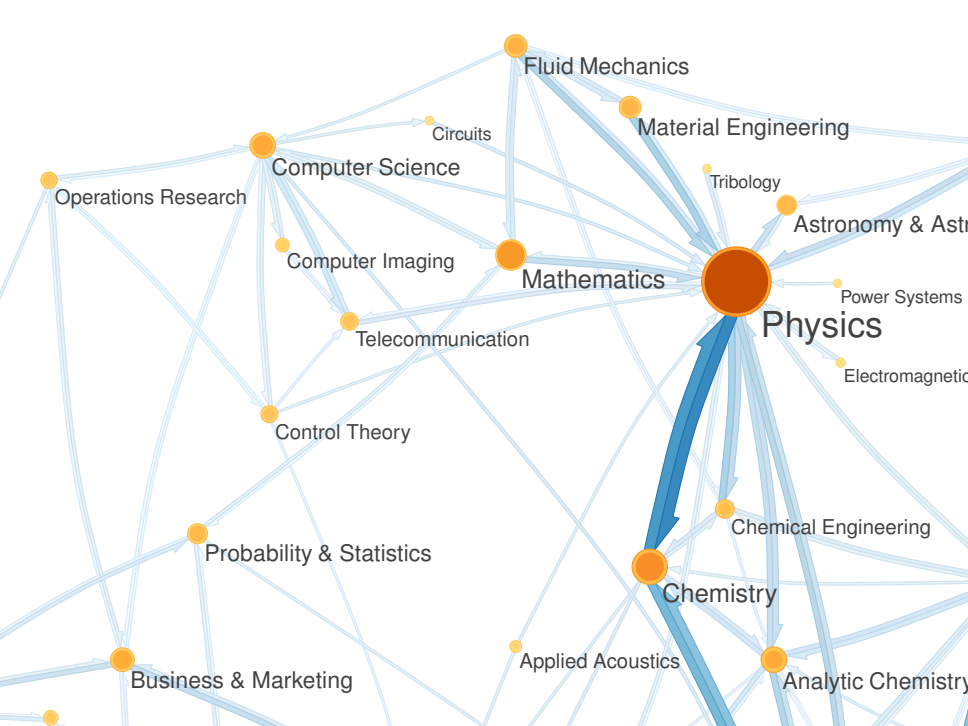
Science 2010

10,000 journals

1,000,000 articles

10,000,000 citations

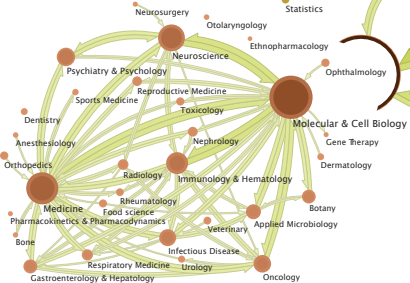
Thomson Scientific Journal Citation Reports
2010



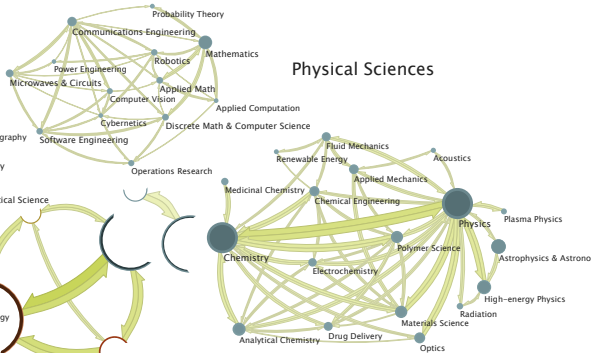
Social Sciences



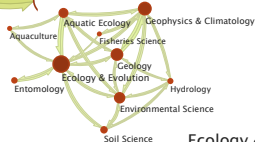
Life Sciences

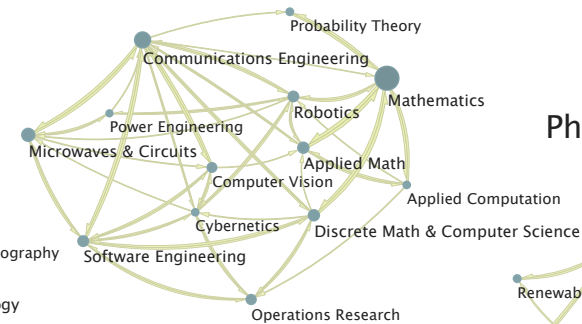


Physical Sciences

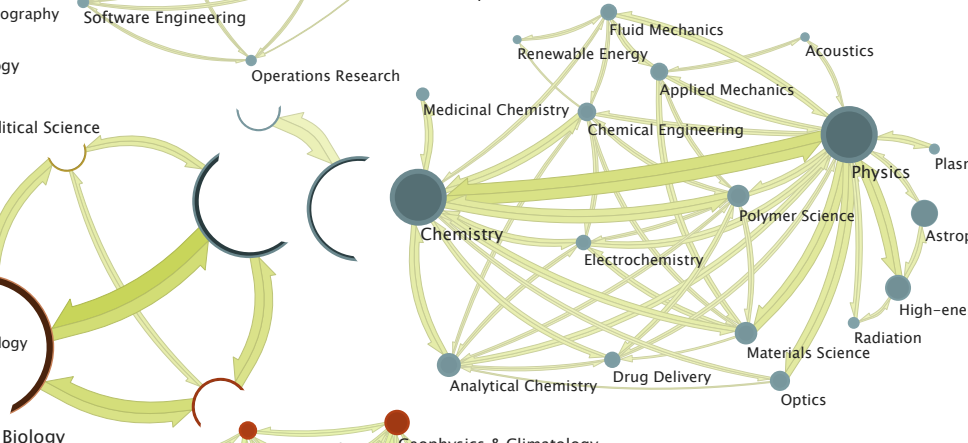


Ecology & Earth Sciences





Physical Sciences



Past

Maps of networks

Carl Bergstrom, Daniel Edler, ...

Now

Memory networks

Renaud Lambiotte, Andrea Lancichinetti, ...

Future

Search and navigation

You?

How are these systems organized?

How are these systems organized
with respect to the flow?

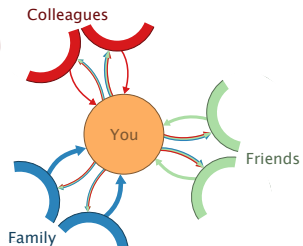
Networks

Networks

describe where flow moves to
depending on where it is

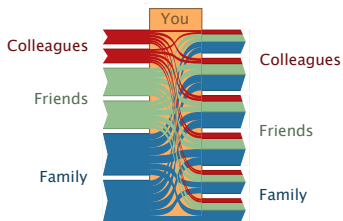
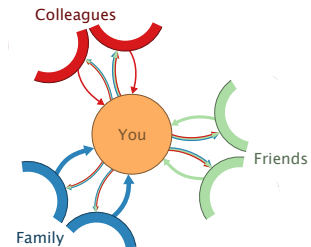
Memoryless dynamics distort constraints on flow

Without memory



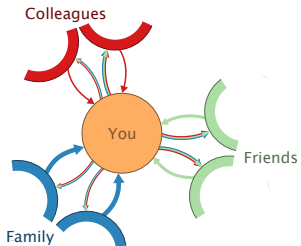
Memoryless dynamics distort constraints on flow

Without memory

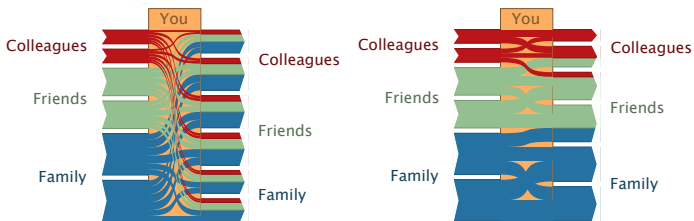


Memoryless dynamics distort constraints on flow

Without memory

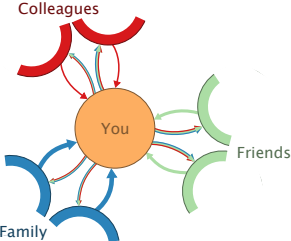


With memory

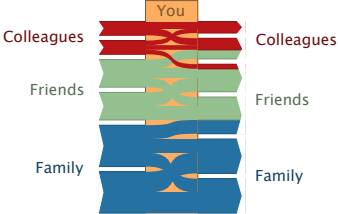
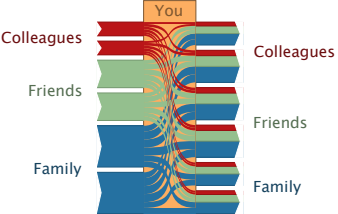
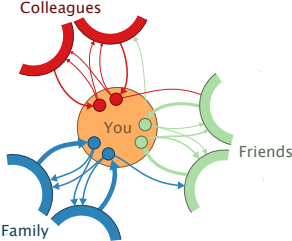


Memoryless dynamics distort constraints on flow

Without memory



With memory



Memory networks

Memory networks

describe where flow moves to
depending on where it comes from

From pathways to networks with and without memory

A

Assemble data

Itinerary	Number of tickets
New York → Chicago → New York	49,632
New York → Chicago → San Francisco	1,031
San Francisco → New York → Chicago → San Francisco	120
Atlanta → Chicago → Atlanta	17,207
Jacksonville → Atlanta → Chicago → Atlanta → Jacksonville	418
⋮	⋮

From pathways to networks with and without memory

A

Assemble data

Itinerary	Number of tickets
New York → Chicago → New York	49,632
New York → Chicago → San Francisco	1,031
San Francisco → New York → Chicago → San Francisco	120
Atlanta → Chicago → Atlanta	17,207
Jacksonville → Atlanta → Chicago → Atlanta → Jacksonville	418
⋮	⋮



B

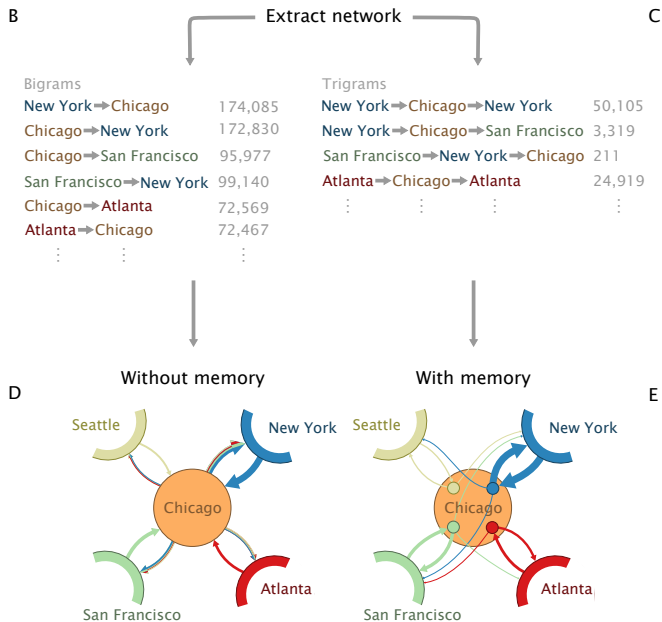
Extract network

C

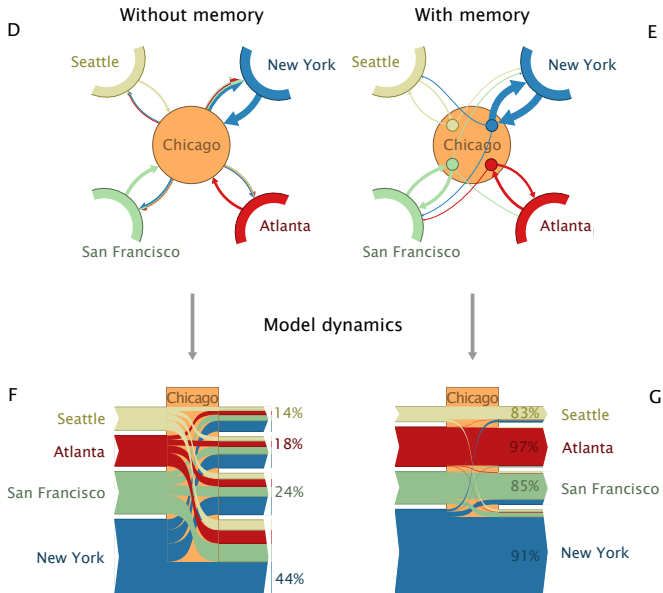
Bigrams	
New York → Chicago	174,085
Chicago → New York	172,830
Chicago → San Francisco	95,977
San Francisco → New York	99,140
Chicago → Atlanta	72,569
Atlanta → Chicago	72,467
⋮	⋮

Trigrams	
New York → Chicago → New York	50,105
New York → Chicago → San Francisco	3,319
San Francisco → New York → Chicago	211
Atlanta → Chicago → Atlanta	24,919
⋮	⋮

From pathways to networks with and without memory



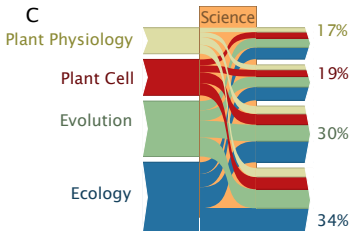
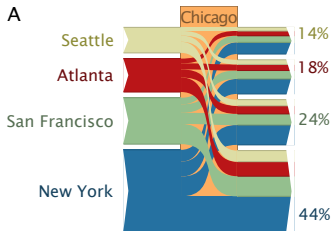
From pathways to networks with and without memory



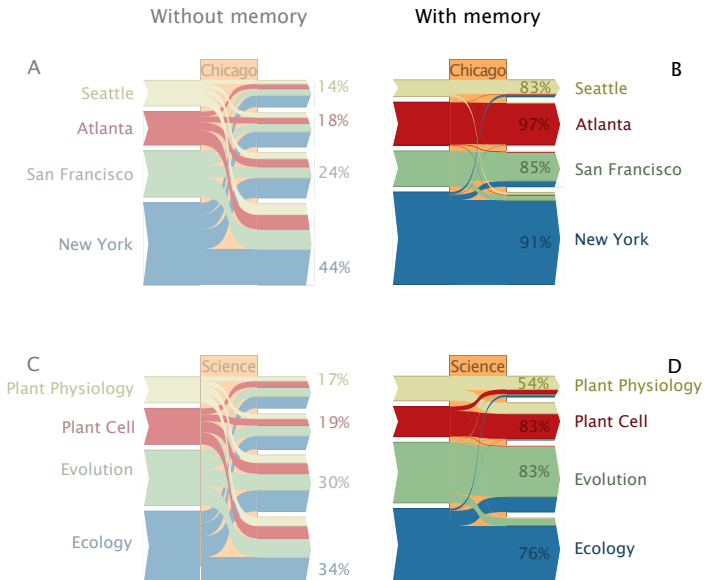


Memoryless dynamics distort real constraints on flow

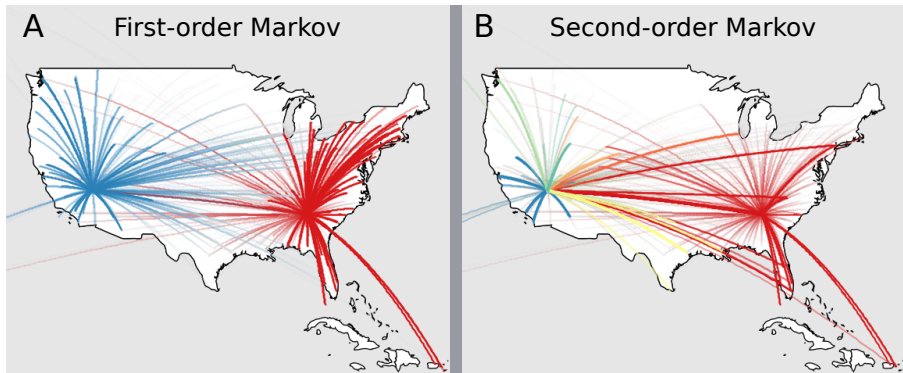
Without memory



Memoryless dynamics distort real constraints on flow



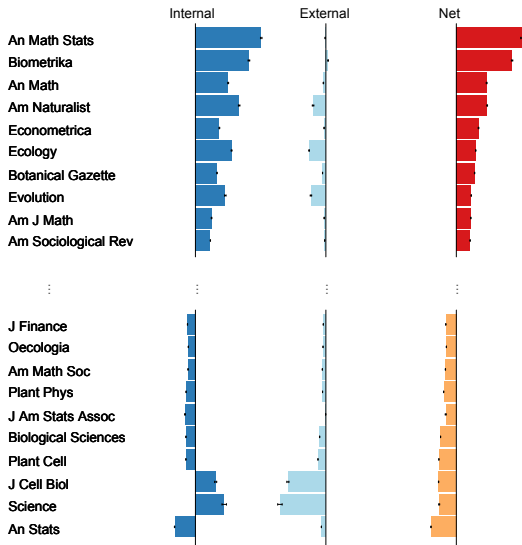
Memory affects clustering of air traffic



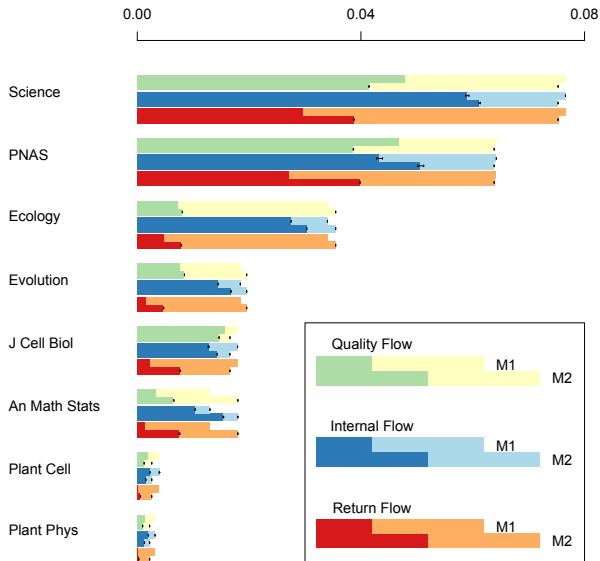
Memory affects clustering of scientific journals

Field	PNAS		Science		Ecology		Plant Cell	
	M1	M2	M1	M2	M1	M2	M1	M2
Ecology	-	■ 13	-	■ 29	■ 100	■ 100	-	-
Cell biology	■ 100	■ 80	■ 100	■ 68	-	-	■ 100	■ 100
Mathematics	-	■ 4.6	-	-	-	-	-	-
Statistics	-	■ 1.5	-	-	-	-	-	-
Anthropology	-	-	-	■ 1.6	-	-	-	-
Others	-	■ 0.38 (1)	-	■ 1.4 (7)	-	-	-	-

Memory affects ranking of scientific journals



Memory affects ranking of scientific journals



Memory matters
because conventional
memoryless dynamics cannot
capture real constraints on flow
with important consequences for
clustering, ranking, and spreading

Past

Maps of networks

Carl Bergstrom, Daniel Edler, ...

Now

Memory networks

Renaud Lambiotte, Andrea Lancichinetti, ...

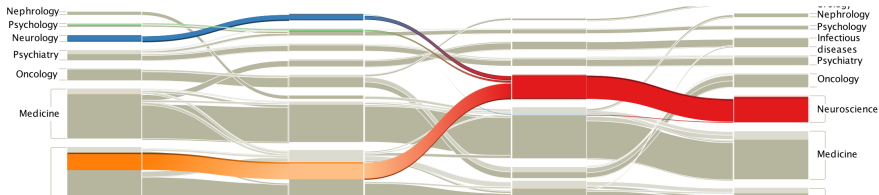
Future

Search and navigation

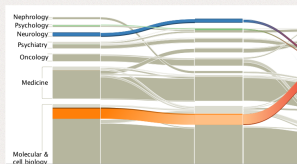
You?

How do we use maps
to improve search and navigation?

Reveal structural change in complex networks



Apps »



Code »

```
using infomath::plogg;
for (unsigned int i = 0; i < numNodes; ++i)
{
  enter_log_enter += plogg(m_moduleFlowData[i].enterFlow);
  exit_log_exit += plogg(m_moduleFlowData[i].exitFlow);
  flow_log_flow += plogg(m_moduleFlowData[i].exitFlow);
  enterFlow += m_moduleFlowData[i].enterFlow;
}
enterFlow += exitNetworkFlow;
enterFlow_log_enterFlow = plogg(enterFlow);
```

Publications »

Maps of information flow reveal community structure in complex networks

Martin Rosvall and Carl T. Bergstrom
PNAS **105**, 11118 (2008). [arXiv:0707.0609]



To comprehend the multipartite organization of large-scale biological and social systems, we introduce a new information-theoretic approach to reveal community structure in