A New Approach towards Detecting Community Structures in Networks

A New Approach towards Detecting Community Structures in Networks

A report based on joint work with

William Y.C.Chen and Winking Q.Yu

from the Center for Combinatorics, LPMC, Nankai University, Tianjin 300071, P.R. China.

Dynamical Systems and Networks

Why are networks currently so popular?

Dynamical Systems and Networks

Why are networks currently so popular?

Networks are **snapshots** of dynamical systems

Dynamical Systems and Networks

Why are networks currently so popular?

Networks are **snapshots** of dynamical systems

and dynamical systems are networks **in action**.

Dynamical Systems and Networks

Why are networks currently so popular?

Networks are **snapshots** of dynamical systems

and dynamical systems are networks **in action**.

Consequently, there is some good hope that proper network analysis can help to elucidate a system's dynamics.

Standard Approaches in Network Analysis

Standard methods of network analysis require a lot of detailed input information about

Standard Approaches in Network Analysis

Standard methods of network analysis require a lot of detailed input information about

the **mechanisms** of interactions between the various agents participating in the network's activity

Standard Approaches in Network Analysis

Standard methods of network analysis require a lot of detailed input information about

the **mechanisms** of interactions between the various agents participating in the network's activity

as well as the respective **inter-and reaction rates**.

Standard Approaches in Network Analysis

Standard methods of network analysis require a lot of detailed input information about

the **mechanisms** of interactions between the various agents participating in the network's activity

as well as the respective **inter-and reaction rates**.

Given such information, a lot of detailed information about the dynamics of actual processes can then be deduced by solving the resulting (ordinary and/or partial differential) equations or by mimicking the interaction schemes by computer simulation.

The Limitations of this Approach

However, in many of the networks currently of interest in biology, such input information is simply just not available.

The Limitations of this Approach

However, in many of the networks currently of interest in biology, such input information is simply just not available.

So, what can be done if all that is (more or less) known are

The Limitations of this Approach

However, in many of the networks currently of interest in biology, such input information is simply just not available.

So, what can be done if all that is (more or less) known are

— the network's agents represented by a collection V of **nodes**,

The Limitations of this Approach

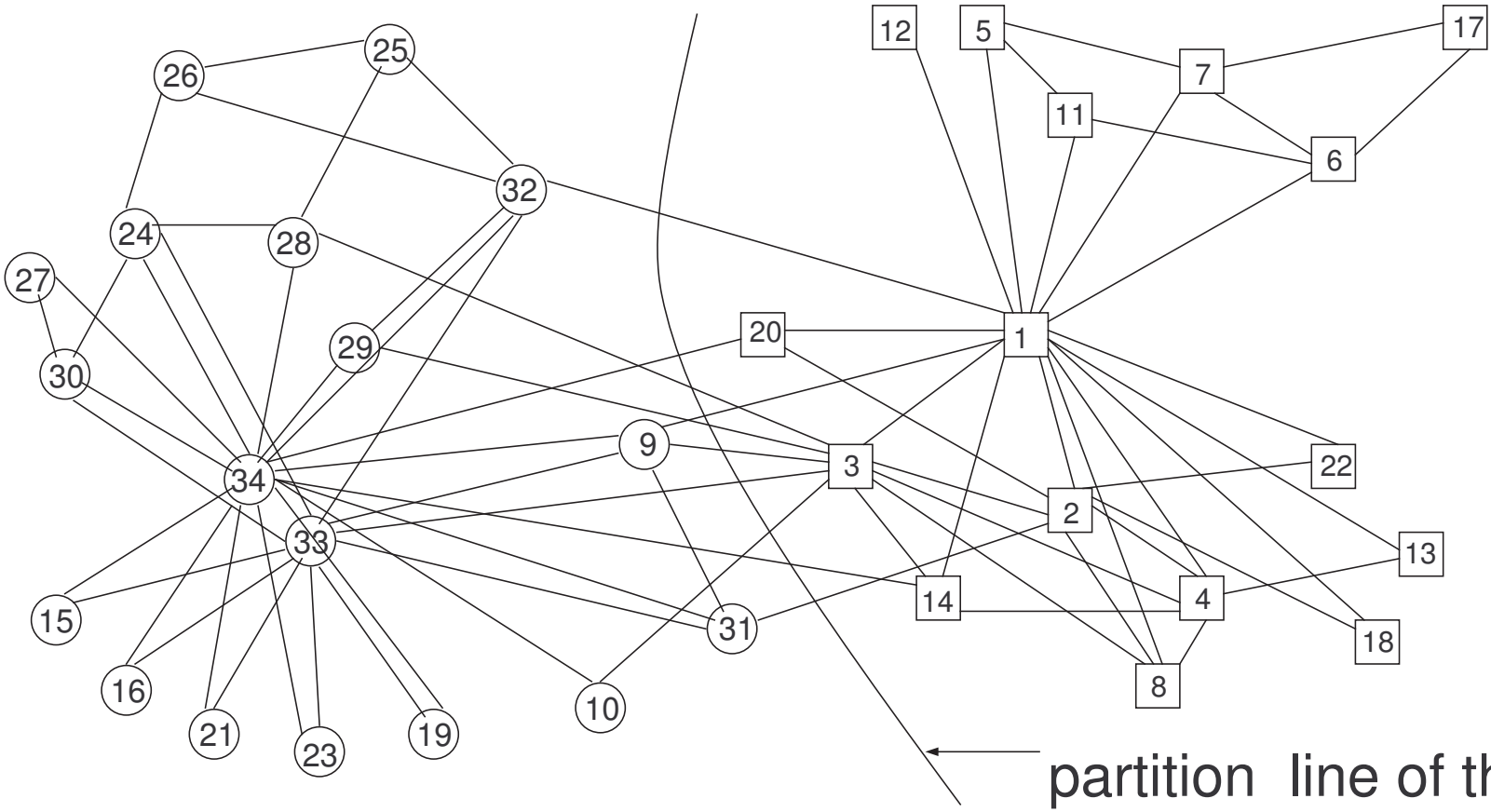
However, in many of the networks currently of interest in biology, such input information is simply just not available.

So, what can be done if all that is (more or less) known are

— the network's agents represented by a collection V of **nodes**,

— and the network's **topology**, i.e., the subset E of the set $\binom{V}{2}$ of all 2-subsets $\{u, v\}$ of V consisting of those pairs of distinct agents u, v that we believe to be closely related to, or to strongly interact with, each other (also called the **edges** of the network)?

Zachary's Karate Club from 1977



Community-Structure Detection

In this talk, I want to report on one currently quite popular proposal to derive some very basic information about the network's agents from nothing but very basic information about the network's topology, that is,

Community-Structure Detection

In this talk, I want to report on one currently quite popular proposal to derive some very basic information about the network's agents from nothing but very basic information about the network's topology, that is,

the proposal to using its topology for deriving its **community structure**,

Community-Structure Detection

In this talk, I want to report on one currently quite popular proposal to derive some very basic information about the network's agents from nothing but very basic information about the network's topology, that is,

the proposal to using its topology for deriving its **community structure**,

or, alternatively phrased, for grouping the network's agents into disjoint **communities** consisting of agents that appear to strongly interact with each other, and not so strongly with the agents in the other communities,

Community-Structure Detection

In this talk, I want to report on one currently quite popular proposal to derive some very basic information about the network's agents from nothing but very basic information about the network's topology, that is,

the proposal to using its topology for deriving its **community structure**,

or, alternatively phrased, for grouping the network's agents into disjoint **communities** consisting of agents that appear to strongly interact with each other, and not so strongly with the agents in the other communities,

with the aim of, e.g., predicting this partition line from the topology of the "friendship" network.

The Current Network Hype

Methods for detecting community structures in networks have received much attention ever since the current network hype began with the proclamation of **scale-free** and **small-world** networks

The Current Network Hype

Methods for detecting community structures in networks have received much attention ever since the current network hype began with the proclamation of **scale-free** and **small-world** networks

> as constituting important new and universally applicable paradigms of interaction schemes observed in real-world systems,

The Current Network Hype

Methods for detecting community structures in networks have received much attention ever since the current network hype began with the proclamation of **scale-free** and **small-world** networks

> as constituting important new and universally applicable paradigms of interaction schemes observed in real-world systems,
>
> and suggesting fundamentally new basic laws governing important processes addressed in the natural and the social sciences.

The Current Network Hype II

According to SCIENCE CITATION INDEX EXPANDED, D.Watts and S.Strogatz' paper on "Collective dynamics of 'small-world' networks" and A.Barabasi and R.Albert's paper on "Emergence of scaling in random networks" both have been quoted more than 1500 times.

The Current Network Hype II

According to SCIENCE CITATION INDEX EXPANDED, D.Watts and S.Strogatz' paper on "Collective dynamics of 'small-world' networks" and A.Barabasi and R.Albert's paper on "Emergence of scaling in random networks" both have been quoted more than 1500 times.

About 30,000 papers have recently mentioned the term NETWORK in their title,

The Current Network Hype II

According to SCIENCE CITATION INDEX EXPANDED, D.Watts and S.Strogatz' paper on "Collective dynamics of 'small-world' networks" and A.Barabasi and R.Albert's paper on "Emergence of scaling in random networks" both have been quoted more than 1500 times.

About 30,000 papers have recently mentioned the term NETWORK in their title,

and more than 100,000 papers have mentioned this term in their abstract.

The Current Network Hype III

And networks play indeed an important role in many fields,

The Current Network Hype III

And networks play indeed an important role in many fields,

from the analysis of the **World-Wide Web**

The Current Network Hype III

And networks play indeed an important role in many fields,

from the analysis of the **World-Wide Web**

to social research, e.g., the analysis of
scientific-collaboration or **citation networks**,

The Current Network Hype III

And networks play indeed an important role in many fields,

from the analysis of the **World-Wide Web**

to social research, e.g., the analysis of **scientific-collaboration** or **citation networks**,

to the life sciences, e.g., the analysis of **ecological, genetic regulatory, protein**, or **metabolic networks**.

Back to Network Communities

As mentioned already, the topic we will be concerned with in this lecture is the **community structure** of networks,

Back to Network Communities

As mentioned already, the topic we will be concerned with in this lecture is the **community structure** of networks,

where the term **community structure** is used here to refer to any partition of the node set V of a given network into a disjoint union of **network modules**,

Back to Network Communities

As mentioned already, the topic we will be concerned with in this lecture is the **community structure** of networks,

where the term **community structure** is used here to refer to any partition of the node set V of a given network into a disjoint union of **network modules**,

i.e., into subsets C of V that have a "higher density" of edges within its **interior** (i.e., among its own nodes) than along its **boundary** (i.e., connecting one of its nodes with a node not contained in C).

Back to Network Communities

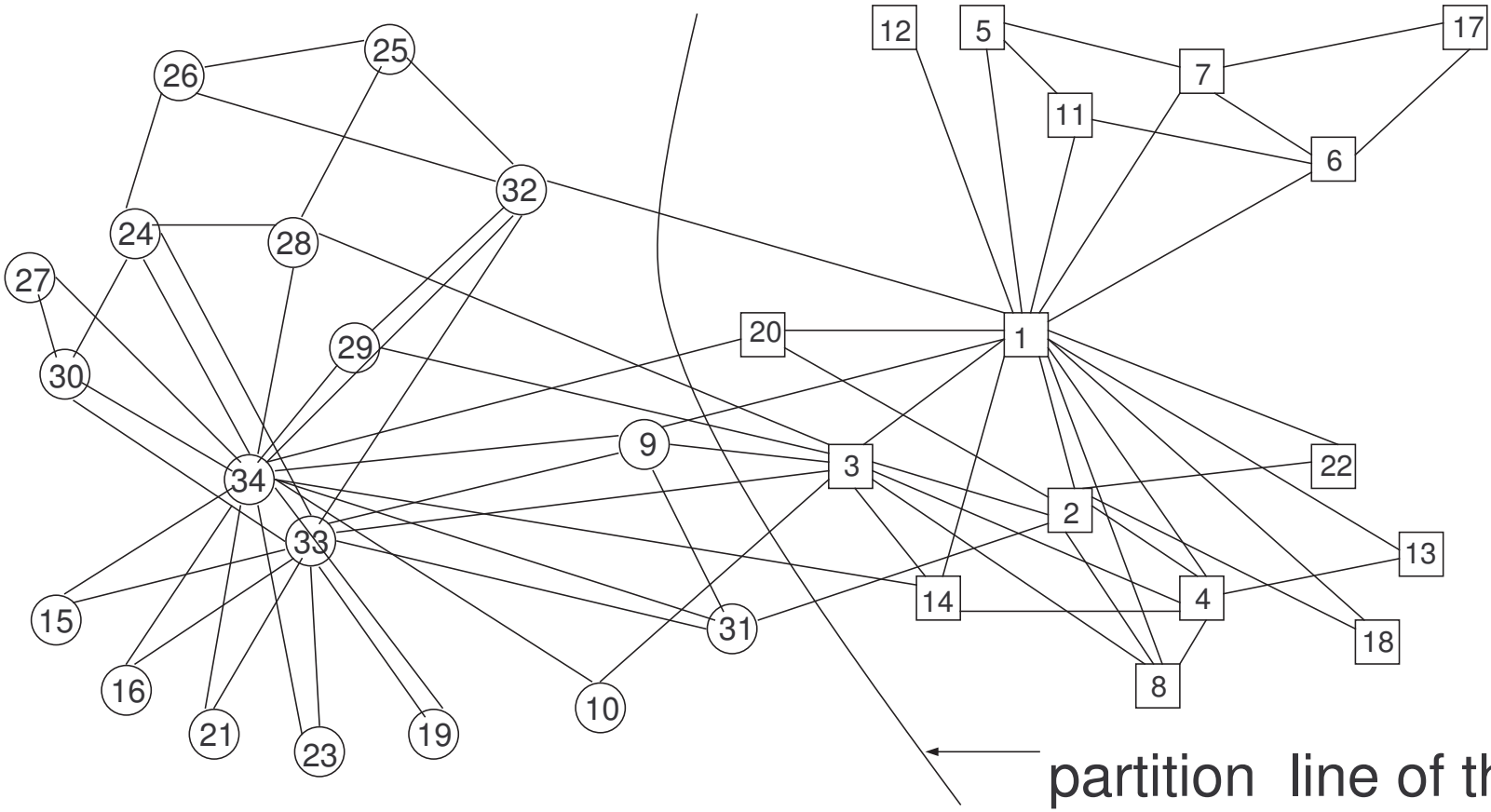
As mentioned already, the topic we will be concerned with in this lecture is the **community structure** of networks,

where the term **community structure** is used here to refer to any partition of the node set V of a given network into a disjoint union of **network modules**,

i.e., into subsets C of V that have a "higher density" of edges within its **interior** (i.e., among its own nodes) than along its **boundary** (i.e., connecting one of its nodes with a node not contained in C).

Here is once again the example of Zachary's Karate Club

Zachary's Famous Karate-Club Example Published in 1977



Recent Approaches

Over the last thirty years, many algorithms were suggested for detecting community structures in networks.

Recent Approaches

Over the last thirty years, many algorithms were suggested for detecting community structures in networks.

A new approach was recently proposed by **Michele Girvan** and **Mark Newman** from the Santa Fe Institute in their 2002 PNAS paper

Recent Approaches

Over the last thirty years, many algorithms were suggested for detecting community structures in networks.

A new approach was recently proposed by **Michele Girvan** and **Mark Newman** from the Santa Fe Institute in their 2002 PNAS paper

Community structure in social and biological networks

Recent Approaches II

resulting in a flurry of many further papers by, e.g.,

- **J.R.Tyler**, **D.M.Wilkinson**, and **B.A.Huberman**,
- **F.Wu** and **B.A.Huberman**,
- **F.Radicchi**, **C. Castellano**, **F. Cecconi**, **V. Loreto**, and **D. Parisi**

and many more.

Then, **M.Newman** and **M.Girvan** introduced the **modularity** parameter to quantify how well a given community structure fits a given network that led to

- a first modularity-optimization algorithm
- that was later improved by the "CNM-algorithm" developed by **A.Clauset**, **M.Newman**, and **C. Moore** and tries to find community structures of high modularity by **greedy** optimization.
- And more work was done by **J.Reichardt** and **S. Bornholdt**, **J.P.Bagrow** and **E.M.Bollt**, and **A.Clauset** again who developed a method for finding "local" community structures.

A Strategy from 1989

Remarkably, a very straight-forward approach towards community-structure detection based on work published by **M.Grötschel** and **Y.Wakabayashi** in 1989 was completely ignored in this context:

A Strategy from 1989

Remarkably, a very straight-forward approach towards community-structure detection based on work published by **M.Grötschel** and **Y.Wakabayashi** in 1989 was completely ignored in this context:

Observe that

A Strategy from 1989

Remarkably, a very straight-forward approach towards community-structure detection based on work published by **M.Grötschel** and **Y.Wakabayashi** in 1989 was completely ignored in this context:

Observe that

> identifying a "community structure" in a network is nothing but inserting and eliminating edges in a somehow **most parsimonious** way so that the network becomes a **disjoint union of cliques** (i.e., "complete subgraphs"),

A Strategy from 1989

Remarkably, a very straight-forward approach towards community-structure detection based on work published by **M.Grötschel** and **Y.Wakabayashi** in 1989 was completely ignored in this context:

Observe that

> identifying a "community structure" in a network is nothing but inserting and eliminating edges in a somehow **most parsimonious** way so that the network becomes a **disjoint union of cliques** (i.e., "complete subgraphs"),
>
> and that such networks are characterized by the **strictly local** property that any two vertices that are connected to a common neighbour are themselves connected by an edge.

A Strategy from 1989

Remarkably, a very straight-forward approach towards community-structure detection based on work published by **M.Grötschel** and **Y.Wakabayashi** in 1989 was completely ignored in this context:

Observe that

> identifying a "community structure" in a network is nothing but inserting and eliminating edges in a somehow **most parsimonious** way so that the network becomes a **disjoint union of cliques** (i.e., "complete subgraphs"),
>
> and that such networks are characterized by the **strictly local** property that any two vertices that are connected to a common neighbour are themselves connected by an edge.

For short, any such network will henceforth be dubbed a **target network**.

A Strategy from 1989

Remarkably, a very straight-forward approach towards community-structure detection based on work published by **M.Grötschel** and **Y.Wakabayashi** in 1989 was completely ignored in this context:

Observe that

> identifying a "community structure" in a network is nothing but inserting and eliminating edges in a somehow **most parsimonious** way so that the network becomes a **disjoint union of cliques** (i.e., "complete subgraphs"),
>
> and that such networks are characterized by the **strictly local** property that any two vertices that are connected to a common neighbour are themselves connected by an edge.

For short, any such network will henceforth be dubbed a **target network**.

Clearly, there is a canonical one-to-one correspondence between **target networks** with node set V on the one, and **set partitions** of V on the other hand.

The Indicator Function of Target Networks

In consequence, using a standard book-keeping device and describing the edges of a graph $G = (V, E)$ with node set V and edge set $E \subseteq \binom{V}{2}$ in terms of the associated **indicator function**

$$\chi_G : \binom{V}{2} \to \{0, 1\} : \{u, v\} \mapsto \chi_G(uv) := \begin{cases} 1 & \text{if } \{u, v\} \in E, \\ 0 & \text{else,} \end{cases}$$

The Indicator Function of Target Networks

In consequence, using a standard book-keeping device and describing the edges of a graph $G = (V, E)$ with node set V and edge set $E \subseteq \binom{V}{2}$ in terms of the associated **indicator function**

$$\chi_G : \binom{V}{2} \to \{0, 1\} : \{u, v\} \mapsto \chi_G(uv) := \begin{cases} 1 & \text{if } \{u, v\} \in E, \\ 0 & \text{else,} \end{cases}$$

a network $T = (V, F)$ is easily seen to be a target network if and only if the linear inequality

$$\chi_T(uv) + \chi_T(vw) - \chi_T(uw) \leq 1$$

is satisfied, for any three distinct nodes $u, v, w \in V$, by its indicator function χ_T.

The Indicator Function of Target Networks II

In other words, using the indicator function allows us to simply reformulate a geometric-combinatorial fact in purely algebraic-numerical terms.

How to Append and to Eliminate Edges in a Most Parsimonious Way to Obtain a Target Network?

Consequently, all that still needs to be done is to find out how, given a network $G = (V, E)$ as above, we can insert and eliminate edges in a **most parsimonious** way so that the resulting network becomes a target network.

How to Append and to Eliminate Edges in a Most Parsimonious Way to Obtain a Target Network?

Consequently, all that still needs to be done is to find out how, given a network $G = (V, E)$ as above, we can insert and eliminate edges in a **most parsimonious** way so that the resulting network becomes a target network.

The most simple way to measure the deviation of the original network $G = (V, E)$ from a given target network $T = (V, F)$ is, of course, the total number of **switched** (i.e., of inserted or eliminated) edges, a number that is easily seen to coincide with the term

How to Append and to Eliminate Edges in a Most Parsimonious Way to Obtain a Target Network?

Consequently, all that still needs to be done is to find out how, given a network $G = (V, E)$ as above, we can insert and eliminate edges in a **most parsimonious** way so that the resulting network becomes a target network.

The most simple way to measure the deviation of the original network $G = (V, E)$ from a given target network $T = (V, F)$ is, of course, the total number of **switched** (i.e., of inserted or eliminated) edges, a number that is easily seen to coincide with the term

$$\sum_{\{u,v\}\in E} \left(1 - \chi_T(uv)\right) + \sum_{\{u,v\}\in \binom{V}{2} - E} \chi_T(uv)$$

How to Append and to Eliminate Edges in a Most Parsimonious Way to Obtain a Target Network?

Consequently, all that still needs to be done is to find out how, given a network $G = (V, E)$ as above, we can insert and eliminate edges in a **most parsimonious** way so that the resulting network becomes a target network.

The most simple way to measure the deviation of the original network $G = (V, E)$ from a given target network $T = (V, F)$ is, of course, the total number of **switched** (i.e., of inserted or eliminated) edges, a number that is easily seen to coincide with the term

$$\sum_{\{u,v\}\in E} \big(1 - \chi_T(uv)\big) + \sum_{\{u,v\}\in \binom{V}{2} - E} \chi_T(uv)$$

giving rise to a **penalty function** that is apparently an affine linear function of the indicator function χ_T.

The Resulting Integer Linear Programming Problem

So, following the approach worked out so excellently by **M.Grötschel** and **Y.Wakabayashi**, we can use integer linear programming (ILP) to find (the indicator function of) an optimal target network relative to that penalty function.

Variations

However, ILP can also easily accommodate much more complex penalty functions:

Variations

However, ILP can also easily accommodate much more complex penalty functions:

We are allowed to specify, for every 2-subset $\{u, v\} \in \binom{V}{2}$ of V, an arbitrary positive or negative number $L_{\textbf{a priori}}(uv)$ registering an **a priori** measure for the likelihood of the pair u, v being contained in the same community within the community structure we want to detect,

Variations

However, ILP can also easily accommodate much more complex penalty functions:

We are allowed to specify, for every 2-subset $\{u, v\} \in \binom{V}{2}$ of V, an arbitrary positive or negative number $L_{\textbf{a priori}}(uv)$ registering an **a priori** measure for the likelihood of the pair u, v being contained in the same community within the community structure we want to detect,

and then use ILP to determine that target network T whose indicator function minimizes the resulting objective function

$$L(T) := \sum_{\{u,v\} \in \binom{V}{2}} \chi_T(uv) L_{\textbf{a priori}}(uv).$$

Variations II

Note that the numbers $L_{\textbf{a priori}}(uv)$ could be derived from the overall local or global graph structure as well as from any additional information we may have been provided with.

Variations II

Note that the numbers $L_{\textbf{a priori}}(uv)$ could be derived from the overall local or global graph structure as well as from any additional information we may have been provided with.

In particular, it may be tempting to experiment with the various edge parameters used in the work by M.Newman and others referred to above.

Our Current 'Ansatz'

Currently, we are using the "CPLEX" software package to investigate this approach,

Our Current 'Ansatz'

Currently, we are using the "CPLEX" software package to investigate this approach,

experimenting, just for a start, with a parameterized **a priori** likelihood function of the form

$$L_{\textbf{a priori}}(uv) := \begin{cases} -\mathbf{s}\left(\deg_G(u) + \deg_G(v)\right) & \text{if } \{u, v\} \in E, \\ 2(|V| - 1) - \deg_G(u) - \deg_G(v) & \text{else.} \end{cases}$$

Our Current 'Ansatz'

Currently, we are using the "CPLEX" software package to investigate this approach,

experimenting, just for a start, with a parameterized **a priori** likelihood function of the form

$$L_{\textbf{a priori}}(uv) := \begin{cases} -\mathbf{s}\left(\deg_G(u) + \deg_G(v)\right) & \text{if } \{u, v\} \in E, \\ 2(|V| - 1) - \deg_G(u) - \deg_G(v) & \text{else.} \end{cases}$$

Here, $\deg_G(x)$ is of course, for any node x in a graph $G = (V, E)$, the number of edges that are incident with it,

Our Current 'Ansatz'

Currently, we are using the "CPLEX" software package to investigate this approach,

experimenting, just for a start, with a parameterized **a priori** likelihood function of the form

$$L_{\textbf{a priori}}(uv) := \begin{cases} -\mathbf{s}\,\big(\deg_G(u) + \deg_G(v)\big) & \text{if } \{u,v\} \in E, \\ 2(|V|-1) - \deg_G(u) - \deg_G(v) & \text{else.} \end{cases}$$

Here, $\deg_G(x)$ is of course, for any node x in a graph $G = (V, E)$, the number of edges that are incident with it,

and **s** is a positive real number that we use for appropriately **calibrating** our objective function.

An Unexpected 'Phase Transition'

Remarkably, increasing the control parameter **s** from 1 to larger and larger values, the running time of the ILP problem becomes shorter and shorter until a value, say, $\mathbf{s}^*$ is found for which

An Unexpected 'Phase Transition'

Remarkably, increasing the control parameter **s** from 1 to larger and larger values, the running time of the ILP problem becomes shorter and shorter until a value, say, $\mathbf{s}^*$ is found for which

- the running time of the associated ILP problem is approximately that of the corresponding **relaxed** LP problem,

An Unexpected 'Phase Transition'

Remarkably, increasing the control parameter **s** from 1 to larger and larger values, the running time of the ILP problem becomes shorter and shorter until a value, say, $\mathbf{s}^*$ is found for which

- the running time of the associated ILP problem is approximately that of the corresponding **relaxed** LP problem,
- the solutions of both problems coincide (i.e., the relaxed problem has an integral solution),

An Unexpected 'Phase Transition'

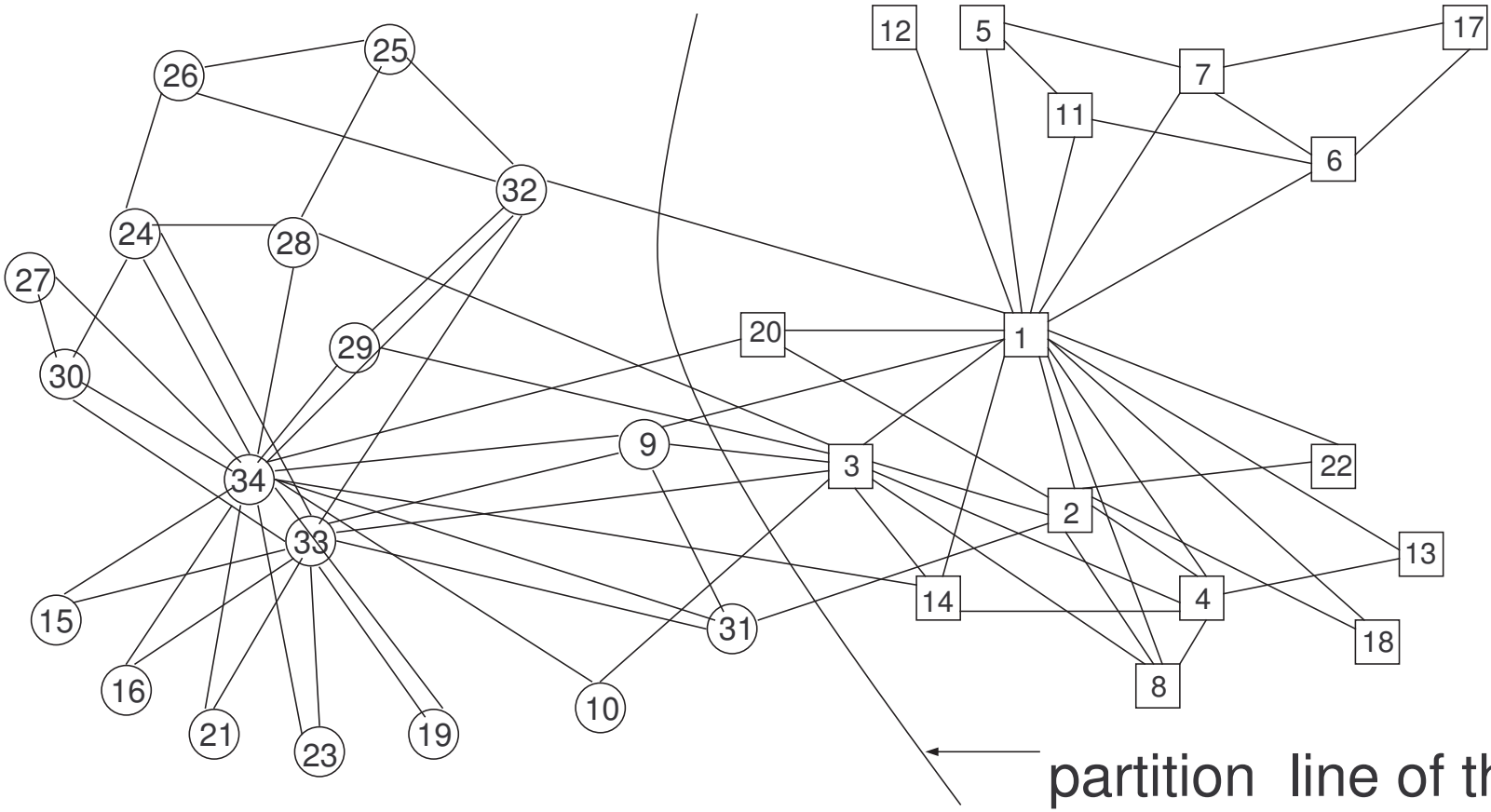
Remarkably, increasing the control parameter **s** from 1 to larger and larger values, the running time of the ILP problem becomes shorter and shorter until a value, say, $\mathbf{s}^*$ is found for which

- the running time of the associated ILP problem is approximately that of the corresponding **relaxed** LP problem,
- the solutions of both problems coincide (i.e., the relaxed problem has an integral solution),
- and the community structure resulting in case $\mathbf{s} = \mathbf{s}^*$ has, so far, consistently turned out to basically coincide with **that** community structure which is considered to be the "correct" one by other researchers.

An Unexpected 'Phase Transition' II

To give an example, let us go back to the well known data regarding "Zachary's Karate Club" that is specified in terms of a simple graph with 34 nodes:

The Famous Karate-Club Example



Results

The (final) result of our algorithm reproduces exactly the same partition line that represents the real-world situation.

Results

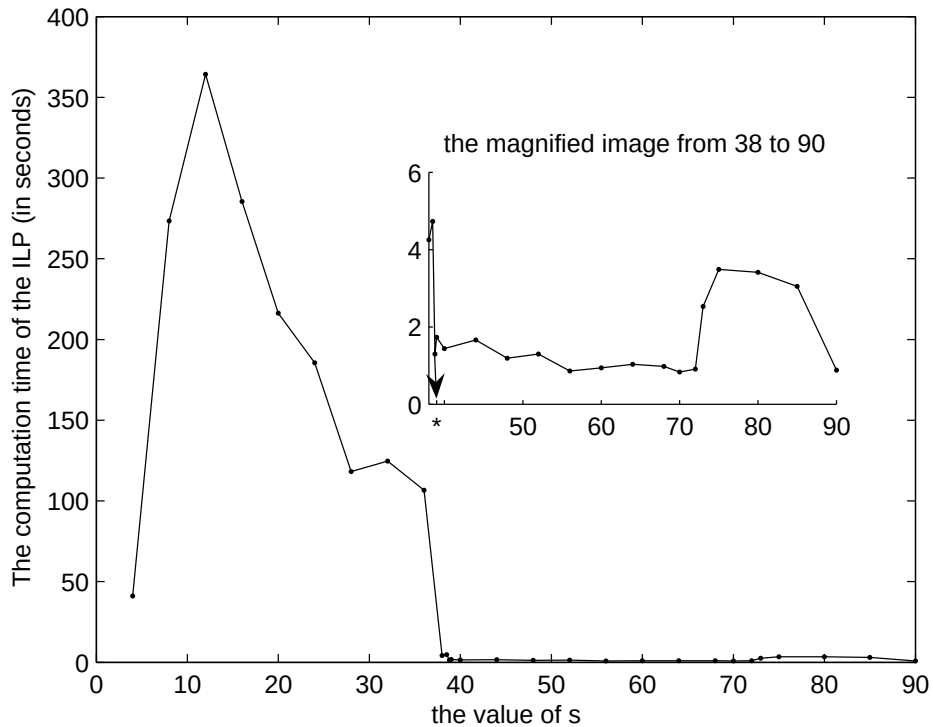
The (final) result of our algorithm reproduces exactly the same partition line that represents the real-world situation.

And it results at $s^* := 38.8$.

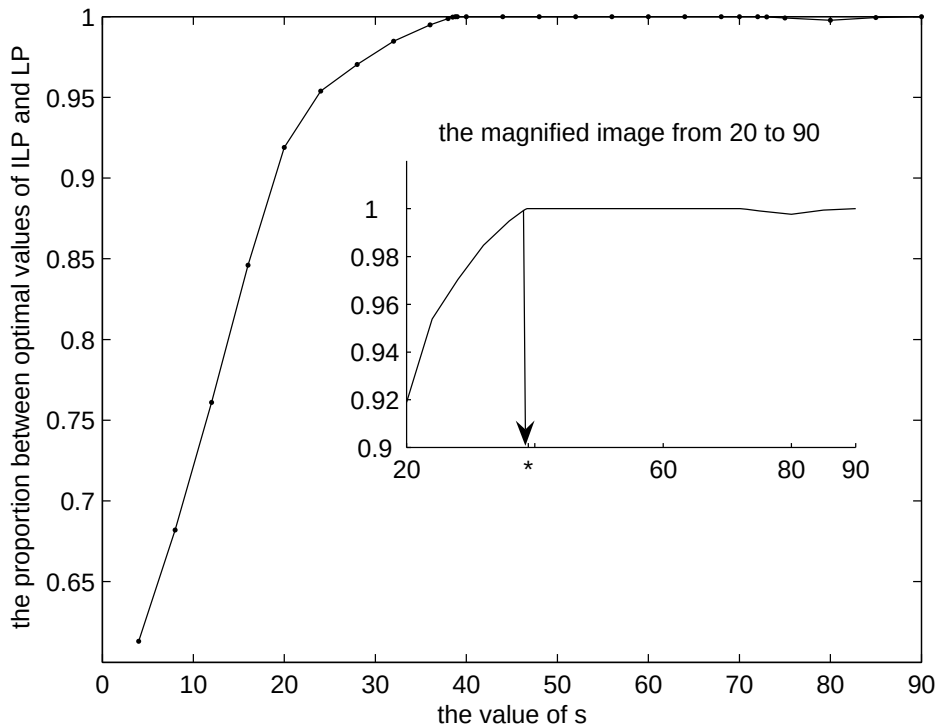
The next diagrams show, as functions of $\mathbf{s}$,

- the time CPLEX needs to find a solution,
- the proportion between the optimal value of the ILP problem and the associated "relaxed" LP problem,
- and that between the respective computation times.

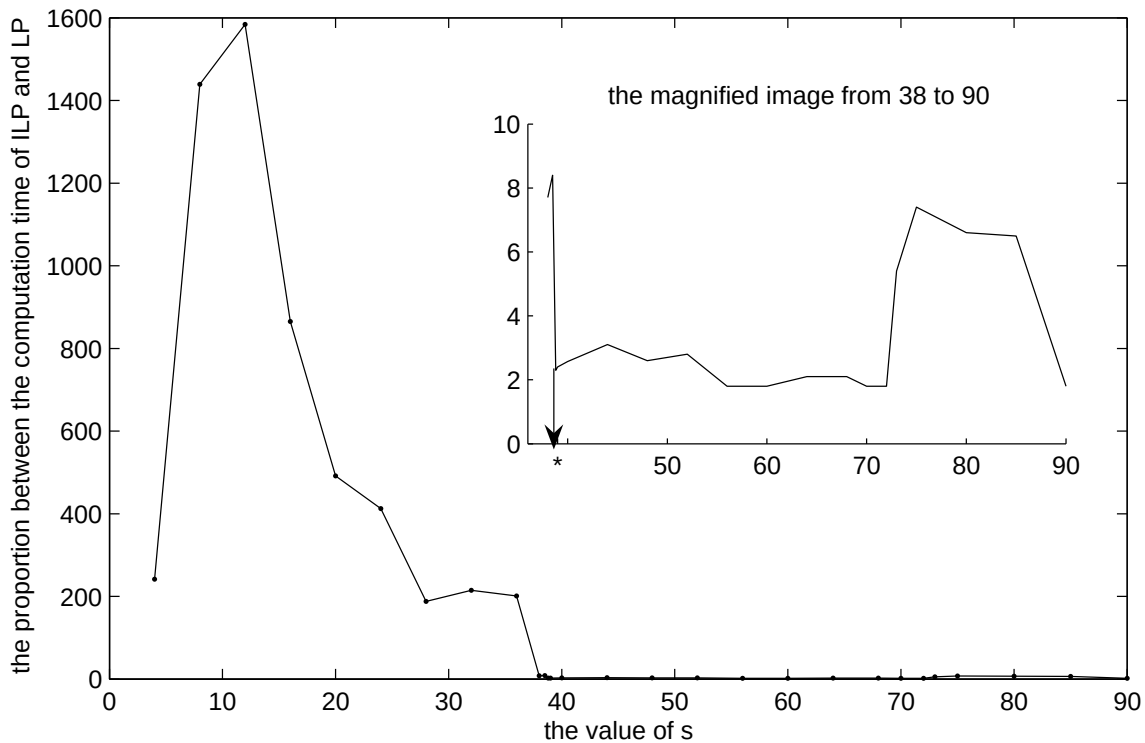
The Computation Time as a Function of **s**



The ILP/LP Objective Function Ratio as a Function of **s**



The ILP/LP Computation Time Ratio as a Function of **s**



Improving the Speed of our Algorithm

In consequence, we propose now to use the following variant of our algorithm:

Improving the Speed of our Algorithm

In consequence, we propose now to use the following variant of our algorithm:

Starting with $s := 1$, run the relaxed LP program for larger and larger values of s until, for the first time, you'll find an integer solution.

Improving the Speed of our Algorithm

In consequence, we propose now to use the following variant of our algorithm:

Starting with $s := 1$, run the relaxed LP program for larger and larger values of s until, for the first time, you'll find an integer solution.

And accept the associated target graph as **the** community structure you search for.

The Chesapeake Bay Food Web

Chesapeake Bay is a large estuary on the east coast of the United States.

The Chesapeake Bay Food Web

Chesapeake Bay is a large estuary on the east coast of the United States.

The **Chesapeake Bay Food Web** relating marine organisms living in this bay was first compiled by **D.Baird** and **R.E.Ulanowicz** referring to those 33 taxa that represent the ecosystem's most prominent species, from phytoplankton and heterotrophic microflagellates to oysters, herring, white perch, blue fish, and striped bass.

The Chesapeake Bay Food Web

Chesapeake Bay is a large estuary on the east coast of the United States.

The **Chesapeake Bay Food Web** relating marine organisms living in this bay was first compiled by **D.Baird** and **R.E.Ulanowicz** referring to those 33 taxa that represent the ecosystem's most prominent species, from phytoplankton and heterotrophic microflagellates to oysters, herring, white perch, blue fish, and striped bass.

The edges indicate **trophic** relationships (i.e., who eats whom).

The Chesapeake Bay Food Web

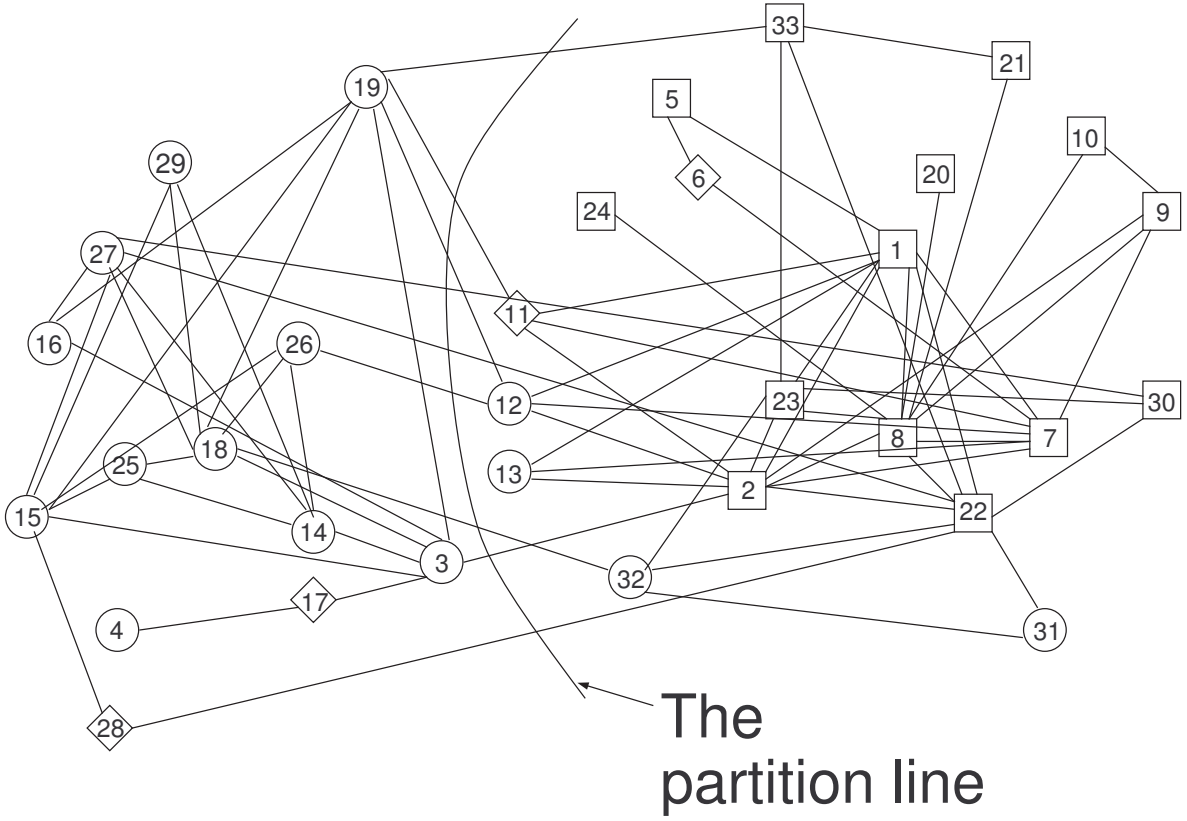
Chesapeake Bay is a large estuary on the east coast of the United States.

The **Chesapeake Bay Food Web** relating marine organisms living in this bay was first compiled by **D.Baird** and **R.E.Ulanowicz** referring to those 33 taxa that represent the ecosystem's most prominent species, from phytoplankton and heterotrophic microflagellates to oysters, herring, white perch, blue fish, and striped bass.

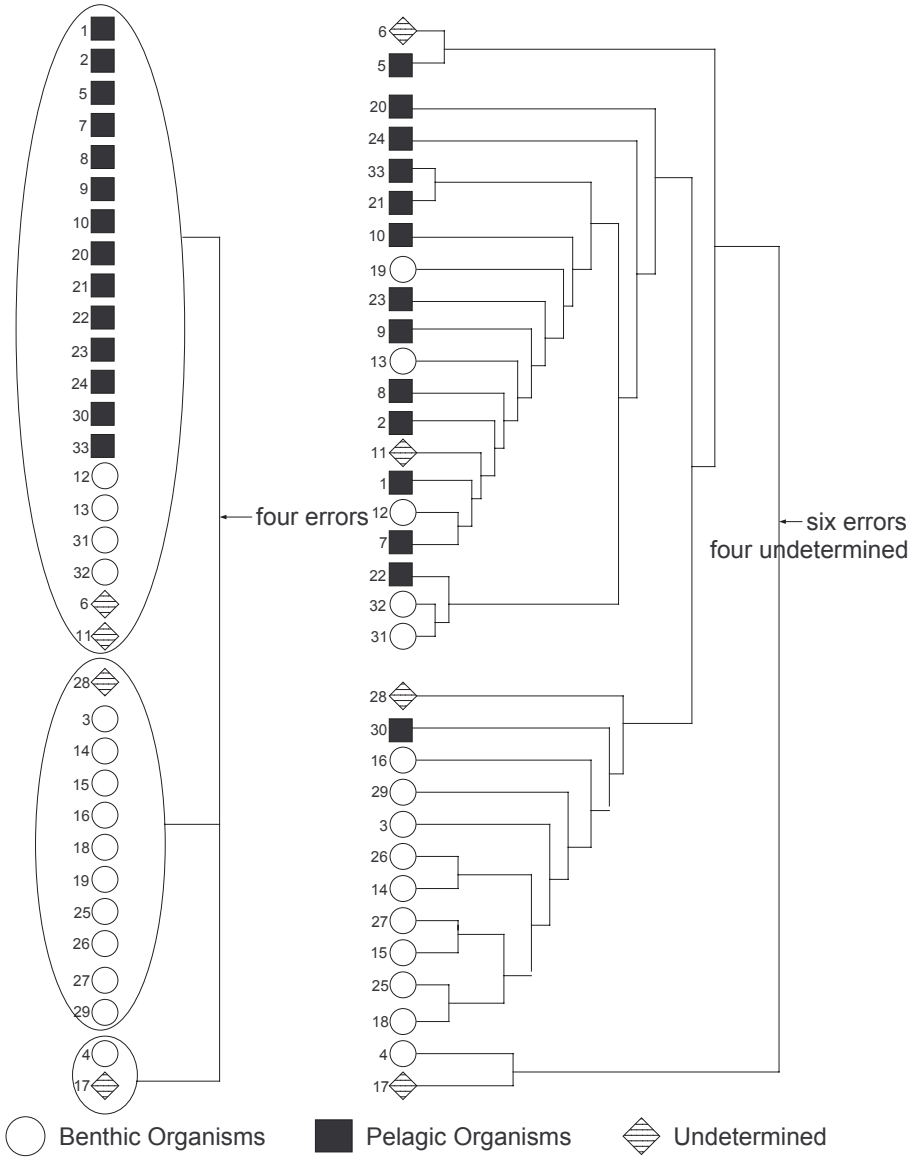
The edges indicate **trophic** relationships (i.e., who eats whom).

We also compare, in the next slide, our result with that of the GN-algorithm.

The Network and the (Principal) Result of our Algorithm



A Comparison with the GN-Algorithm



A Letter from Robert Ulanowicz

Oct. 24, 2006

Dear Andreas and Winking,

A Letter from Robert Ulanowicz

Oct. 24, 2006

Dear Andreas and Winking,

I finally had the chance to look at the groupings that you sent to me. To begin with, I noticed only a single transposition difference between your grouping and that by Girvan and Newman. Namely, you group blue crab (19) in the second group and blue fish (30) in the first. GN does the opposite.

A Letter from Robert Ulanowicz

Oct. 24, 2006

Dear Andreas and Winking,

I finally had the chance to look at the groupings that you sent to me. To begin with, I noticed only a single transposition difference between your grouping and that by Girvan and Newman. Namely, you group blue crab (19) in the second group and blue fish (30) in the first. GN does the opposite.

My judgement is that both groupings are quite good, but yours probably wins out by a hair.

A Letter from Robert Ulanowicz II

Please notice that the groupings are according to where the organisms "feed", not where they are located. Hence, three of the "mistakes" you indicated are not mistakes at all by my reckoning.

A Letter from Robert Ulanowicz II

Please notice that the groupings are according to where the organisms "feed", not where they are located. Hence, three of the "mistakes" you indicated are not mistakes at all by my reckoning.

That mya (12), oysters(13) and other suspension feeders (11) live on the bottom is only incidental. They are all filter feeders and take their nourishment from the water column. In terms of **feeding**, they belong with the pelagic organisms.

A Letter from Robert Ulanowicz II

Please notice that the groupings are according to where the organisms "feed", not where they are located. Hence, three of the "mistakes" you indicated are not mistakes at all by my reckoning.

That mya (12), oysters(13) and other suspension feeders (11) live on the bottom is only incidental. They are all filter feeders and take their nourishment from the water column. In terms of **feeding**, they belong with the pelagic organisms.

On the other hand, spot (27), white perch (28), croaker (25), catfish (29) and hogchoker (26) are technically nektonic, but they all derive their nourishment primarily from the benthos and can logically be placed in the second category.

A Letter from Robert Ulanowicz III

As for the single discrepancy between the two methods, I would judge that blue crabs (19) belong decidedly in the benthic feeding group, as your method detected.

A Letter from Robert Ulanowicz III

As for the single discrepancy between the two methods, I would judge that blue crabs (19) belong decidedly in the benthic feeding group, as your method detected.

Blue fish (30) feed mostly on other nekton, but ultimately derive most of their sustenance from the benthos.

A Letter from Robert Ulanowicz III

As for the single discrepancy between the two methods, I would judge that blue crabs (19) belong decidedly in the benthic feeding group, as your method detected.

Blue fish (30) feed mostly on other nekton, but ultimately derive most of their sustenance from the benthos.

In fact, in the paper I sent you on Oct. 10, we note how the indirect diet of striped bass (33) differs from that of blue fish (30) because the former derives most of its sustenance from the pelagic domain, whereas the latter comes ultimately (but not directly) from the benthos.

A Letter from Robert Ulanowicz IV

Hence, blue fish is a "borderline" species, and GN do not err gravely by placing it among the benthic feeders. Their bigger error is in placing blue crab (19) among the pelagic feeders.

A Letter from Robert Ulanowicz IV

Hence, blue fish is a "borderline" species, and GN do not err gravely by placing it among the benthic feeders. Their bigger error is in placing blue crab (19) among the pelagic feeders.

So by placing blue crab correctly among the benthic feeders, you win by a slight edge. :)

A Letter from Robert Ulanowicz IV

Hence, blue fish is a "borderline" species, and GN do not err gravely by placing it among the benthic feeders. Their bigger error is in placing blue crab (19) among the pelagic feeders.

So by placing blue crab correctly among the benthic feeders, you win by a slight edge. :)

I do hope these observations have been helpful.

A Letter from Robert Ulanowicz IV

Hence, blue fish is a "borderline" species, and GN do not err gravely by placing it among the benthic feeders. Their bigger error is in placing blue crab (19) among the pelagic feeders.

So by placing blue crab correctly among the benthic feeders, you win by a slight edge. :)

I do hope these observations have been helpful.

I am impressed by the power of your grouping algorithm.

A Letter from Robert Ulanowicz IV

Hence, blue fish is a "borderline" species, and GN do not err gravely by placing it among the benthic feeders. Their bigger error is in placing blue crab (19) among the pelagic feeders.

So by placing blue crab correctly among the benthic feeders, you win by a slight edge. :)

I do hope these observations have been helpful.

I am impressed by the power of your grouping algorithm.

Sincerely, Bob

A Perturbation Experiment

Finally, some very recent results that we obtained studying the "reconstructibility" of a given target graph $T = (V, E)$ from graphs G obtained from T by systematically **perturbing** T:

A Perturbation Experiment

Finally, some very recent results that we obtained studying the "reconstructibility" of a given target graph $T = (V, E)$ from graphs G obtained from T by systematically **perturbing** T:

We choose the disjoint union of 4 cliques containing $12, 9, 8, 6$ nodes, respectively, — and, hence, altogether
$66 + 36 + 28 + 15 = 145$ edges – as our target graph T.

A Perturbation Experiment II

To explore how well this graph is reconstructed from the perturbed graphs G, we

A Perturbation Experiment II

To explore how well this graph is reconstructed from the perturbed graphs G, we

(1) considered randomly generated graphs $H = (V, F)$ whose edge sets were, in each instance, supposed to represent all the edges that were to be switched, and formed the graph $G := \big(V, (E - F) \cup (F - E)\big)$,

A Perturbation Experiment II

To explore how well this graph is reconstructed from the perturbed graphs G, we

(1) considered randomly generated graphs $H = (V, F)$ whose edge sets were, in each instance, supposed to represent all the edges that were to be switched, and formed the graph $G := \big(V, (E - F) \cup (F - E)\big)$,

(2) applied our algorithm to G yielding a target graph $T' = (V, E')$,

A Perturbation Experiment II

To explore how well this graph is reconstructed from the perturbed graphs G, we

(1) considered randomly generated graphs $H = (V, F)$ whose edge sets were, in each instance, supposed to represent all the edges that were to be switched, and formed the graph $G := \big(V, (E - F) \cup (F - E)\big)$,

(2) applied our algorithm to G yielding a target graph $T' = (V, E')$,

(3) and considered the resulting **maintenance ratio** relative to H, i.e., the quotient of (i) the (minimal) number of nodes that have to be moved from one clique to another one to obtain T from T' and (ii) the number $|V|$ of all nodes in V (which happens to be 35).

A Perturbation Experiment II

To explore how well this graph is reconstructed from the perturbed graphs G, we

(1) considered randomly generated graphs $H = (V, F)$ whose edge sets were, in each instance, supposed to represent all the edges that were to be switched, and formed the graph $G := \big(V, (E - F) \cup (F - E)\big)$,

(2) applied our algorithm to G yielding a target graph $T' = (V, E')$,

(3) and considered the resulting **maintenance ratio** relative to H, i.e., the quotient of (i) the (minimal) number of nodes that have to be moved from one clique to another one to obtain T from T' and (ii) the number $|V|$ of all nodes in V (which happens to be 35).

The following table lists the average maintenance ratio obtained for (ten) random graphs $H = (V, F)$ for any given number $|F|$ of edges or, equivalently, for any given **perturbation ratio**, that is, the quotient $|F|/|E| = |F|/145$, using in addition the most simple possible objective function defined by putting

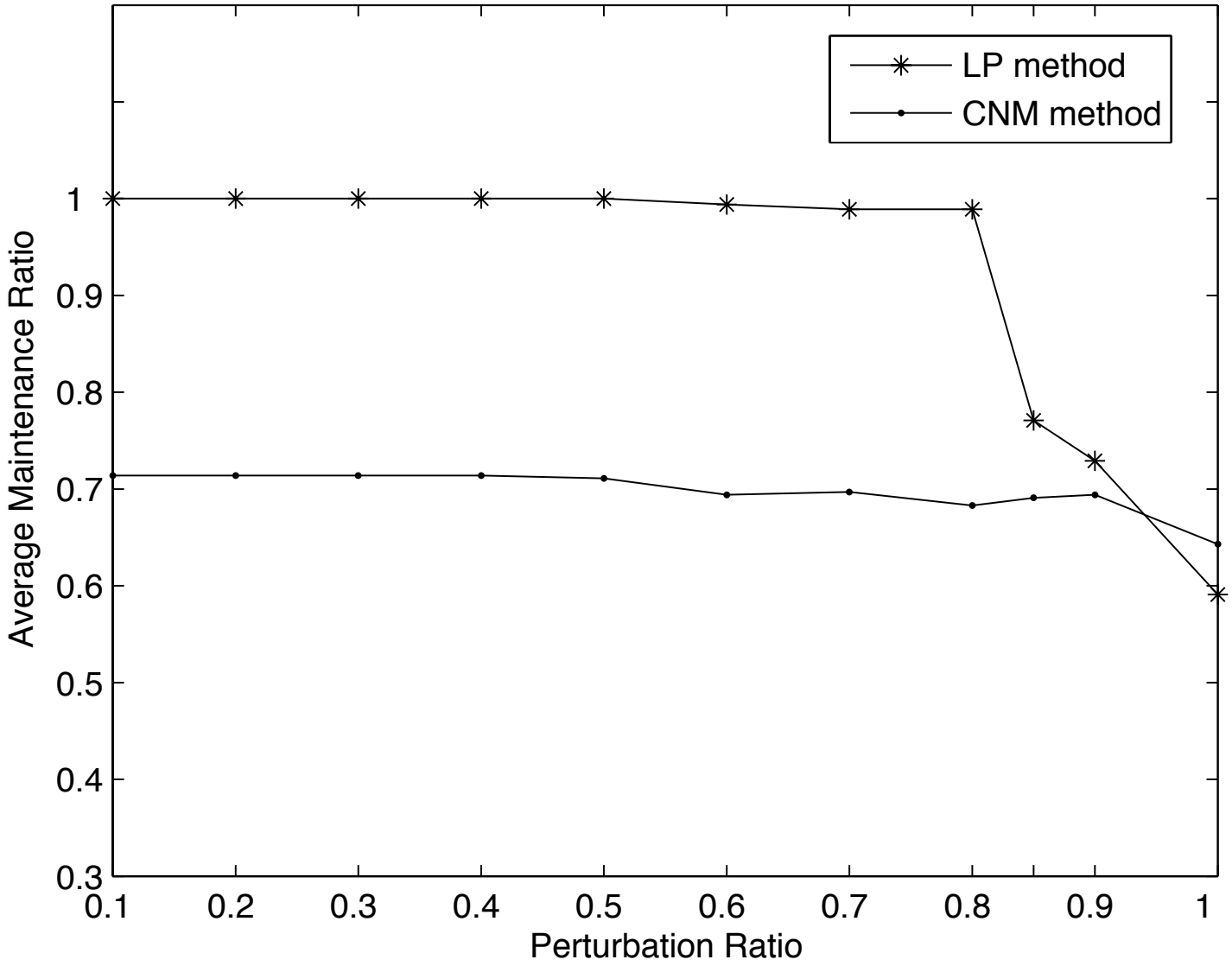
$$L_{\textbf{a priori}}(uv) := \begin{cases} 1 & \text{if } \{u, v\} \notin E, \\ -\mathbf{s} & \text{else .} \end{cases}$$

A Perturbation Experiment III

The perturbation and the maintenance ratio

Number of switched edges	Perturbation ratio	Maintenance ratio
15	0.103	1
29	0.200	1
44	0.303	1
51	0.352	1
53	0.366	1
55	0.379	1
58	0.400	1
73	0.503	1
87	0.600	0.99
102	0.703	1
116	0.800	0.98
131	0.903	0.63
145	1	0.36

A Comparison with the CNM Algorithm



From a Letter from August 27, 2006, by Aaron Clauset, regarding the CNM-algorithm

Dear Andreas,

If your work is going to be presented in the computer science community, it would be most correct to call our approach a heuristic; the physics community is less picky, so there, it would be an algorithm.

From a Letter from August 27, 2006, by Aaron Clauset, regarding the CNM-algorithm

Dear Andreas,

If your work is going to be presented in the computer science community, it would be most correct to call our approach a heuristic; the physics community is less picky, so there, it would be an algorithm.

In either case, it would be most correct to say that it is a very fast greedy approach to the maximum-modularity problem, and that it returns demonstrably good results for many real-world networks.

From a Letter from August 27, 2006, by Aaron Clauset, regarding the CNM-algorithm

Dear Andreas,

If your work is going to be presented in the computer science community, it would be most correct to call our approach a heuristic; the physics community is less picky, so there, it would be an algorithm.

In either case, it would be most correct to say that it is a very fast greedy approach to the maximum-modularity problem, and that it returns demonstrably good results for many real-world networks.

You could also note that it is not an optimal greedy algorithm, and that it is known to return poor results for certain pathological networks that appear to be unlike those we see in the real world. It doesn't really matter how big (or small) the pathological network is, the greedy approach will always give you a bad answer.

From a Letter from August 27, 2006, by Aaron Clauset II

So, the most accurate characterization would be to say that, as the size of the network you want to cluster increases, you have progressively fewer choices of algorithms for maximizing the modularity (or any other of those nice parameters used for community-structure detection) because most of them take time at least $O(n^2)$, and we want our algorithms to return an answer after a reasonable amount of time.

From a Letter from August 27, 2006, by Aaron Clauset II

So, the most accurate characterization would be to say that, as the size of the network you want to cluster increases, you have progressively fewer choices of algorithms for maximizing the modularity (or any other of those nice parameters used for community-structure detection) because most of them take time at least $O(n^2)$, and we want our algorithms to return an answer after a reasonable amount of time.

For the very largest networks, say on the order of hundreds of thousands or millions of nodes, basically only our greedy algorithm will return an answer to you within this constraint.

Directions for Future Work

Our results imply various questions that deserve to be investigated further and suggest several tasks that deserve to be pursued in the future:

Directions for Future Work

Our results imply various questions that deserve to be investigated further and suggest several tasks that deserve to be pursued in the future:

- Optimizing the use of the CPLEX program,

Directions for Future Work

Our results imply various questions that deserve to be investigated further and suggest several tasks that deserve to be pursued in the future:

- Optimizing the use of the CPLEX program,
- studying a larger range of objective functions and trying to determine those that seem to be particularly **appropriate** for a specific task, including objective functions related to edge-weighted networks and asymmetric ones representing directed networks,

Directions for Future Work

Our results imply various questions that deserve to be investigated further and suggest several tasks that deserve to be pursued in the future:

- Optimizing the use of the CPLEX program,
- studying a larger range of objective functions and trying to determine those that seem to be particularly **appropriate** for a specific task, including objective functions related to edge-weighted networks and asymmetric ones representing directed networks,
- trying to understand the influence exercised by, and in particular the apparent "phase transition" behaviour of, the control parameter **s**,

Directions for Future Work

Our results imply various questions that deserve to be investigated further and suggest several tasks that deserve to be pursued in the future:

- Optimizing the use of the CPLEX program,
- studying a larger range of objective functions and trying to determine those that seem to be particularly **appropriate** for a specific task, including objective functions related to edge-weighted networks and asymmetric ones representing directed networks,
- trying to understand the influence exercised by, and in particular the apparent "phase transition" behaviour of, the control parameter **s**,
- analysing the "landscape" defined on the set of target graphs by a given (**s**-parametrized) objective function using stochastic models and, in particular, the entropy concept from statistical physics,

Directions for Future Work

Our results imply various questions that deserve to be investigated further and suggest several tasks that deserve to be pursued in the future:

- Optimizing the use of the CPLEX program,
- studying a larger range of objective functions and trying to determine those that seem to be particularly **appropriate** for a specific task, including objective functions related to edge-weighted networks and asymmetric ones representing directed networks,
- trying to understand the influence exercised by, and in particular the apparent "phase transition" behaviour of, the control parameter **s**,
- analysing the "landscape" defined on the set of target graphs by a given (**s**-parametrized) objective function using stochastic models and, in particular, the entropy concept from statistical physics,
- trying to understand also the apparent "phase transition" behaviour of the maintenance ratio relative to the perturbation ratio,

Directions for Future Work

Our results imply various questions that deserve to be investigated further and suggest several tasks that deserve to be pursued in the future:

- Optimizing the use of the CPLEX program,
- studying a larger range of objective functions and trying to determine those that seem to be particularly **appropriate** for a specific task, including objective functions related to edge-weighted networks and asymmetric ones representing directed networks,
- trying to understand the influence exercised by, and in particular the apparent "phase transition" behaviour of, the control parameter **s**,
- analysing the "landscape" defined on the set of target graphs by a given (**s**-parametrized) objective function using stochastic models and, in particular, the entropy concept from statistical physics,
- trying to understand also the apparent "phase transition" behaviour of the maintenance ratio relative to the perturbation ratio,
- developing approximative algorithms for **large-scale** applications,

Directions for Future Work

Our results imply various questions that deserve to be investigated further and suggest several tasks that deserve to be pursued in the future:

- Optimizing the use of the CPLEX program,
- studying a larger range of objective functions and trying to determine those that seem to be particularly **appropriate** for a specific task, including objective functions related to edge-weighted networks and asymmetric ones representing directed networks,
- trying to understand the influence exercised by, and in particular the apparent "phase transition" behaviour of, the control parameter **s**,
- analysing the "landscape" defined on the set of target graphs by a given (**s**-parametrized) objective function using stochastic models and, in particular, the entropy concept from statistical physics,
- trying to understand also the apparent "phase transition" behaviour of the maintenance ratio relative to the perturbation ratio,
- developing approximative algorithms for **large-scale** applications,
- creating a data base containing the results obtained by applying the algorithm(s) to **real-world** data gathered from the existing literature.

Thanks

Thank You for Your Patience and Attention!

Thanks

Thank You for Your Patience and Attention!

Xie, Xie ! !! !!!

References

R. Albert, H. Jeong, and A.-L. Barabási, Diameter of the World-Wide Web. *Nature* 401, 130-131 (1999).

L. A. N. Amaral, A. Scala, M. Barthélémy and H. E. Stanley, Classes of samll-world networks. *Proc. Natl. Acad. Sci. USA* 97, 11149-11152 (2000).

J. P. Bagrow and E. M. Bollt, Local method for detecting communities. *Phys. Rev. E* 72, 046108 (2005).

D. Baird and R. E. Ulanowicz, The seasonal dynamics of the Chesapeake Bay ecosystem. *Ecological Monographs* 59, 329-364 (1989).

A.-L. Barabási and R. Albert, Emergence of scaling in random networks. *Science* 286, 509-512 (1999).

A. Clauset, Finding local community structure in networks. *Phys. Rev. E* 72, 026132 (2005).

A. Clauset, M. E. J. Newman, and C. Moore, Finding community structure in very large networks, *Phys. Rev. E* 69, 026113 (2004).

E. Davidson et al, A Genomic Regulatory Network for Development. *Science* 295, 1669-1678 (2002).

L. Donetti and M. Muñoz, Detecting network communities: a new systematic and efficient algorithm. *J. Stat. Mech* 10, 10012 (2004).

G. W. Flake, S. R. Lawrence, C. L. Giles, and F. M. Coetzee, Self-organization and identification of Web communities. *IEEE Computer* 35, 66-71 (2002).

M. Girvan and M. E. J. Newman, Community structure in social and biological networks. *Proc. Natl. Acad. Sci. USA* 99, 7821–7826 (2002).

M. Grötschel, Y. Wakabayashi, A cutting plane algorithm for a clustering problem. *Mathematical Programming* 45, 59-96 (1989).

M. Grötschel, Y. Wakabayashi, Facets of the Clique Partitioning Polytope. *Mathematical Programming* 47, 367-387 (1990).

ILOG: ILOG CPLEX 9.1 Users Manual. ILOG, 2005.

H. Jeong, S. P. Mason, A.-L. Barabási and Z. N. Oltvai, Lethality and centrality in protein networks. *Nature* 411, 41-42 (2001).

H. Jeong, B. Tombor, R. Albert, Z. N. Oltvai, and A.-L. Barabási, The large-scale organization of metabolic networks. *Nature* 407, 651-654 (2000).

B. W. Kernighan and S. Lin, An efficient heuristic procedure for partitioning graphs. *Bell System Technical Journal* 49, 291-307 (1970).

J. Kleinberg and S. Lawrence, The structure of the Web. *Science* 294, 1849-1850 (2001).

A. E. Krause, K. A. Frank, D. M. Mason, R. E. Ulanowlcz and W. W. Taylor, Compartments revealed in food-web structures. *Nature* 426, 282-285 (2003).

S. J. McNaughton, Stability and diversity of ecological communities. *Nature* 274, 251-252 (1978).

J. C. Moore and H. W. Hunt, Resource compartmentation and the stability of real ecosystems. *Nature* 333, 261-263 (1988).

G. L . Nemhauser, L. A. Wolsey, *Integer and Combinatorial Optimization*. John Wiley & Sons, Inc., USA (1988).

M. E. Newman, The structure of scientific collaboration networks. *Proc. Natl. Acad. Sci. USA* 98, 404-409 (2001).

M. E. Newman, Fast algorithm for detecting community structure in networks. *Phys. Rev. E* 69, 066133 (2004).

M. E. Newman, Detecting community structure in networks. *Eur. Phys. J. B* 38, 321-330 (2004).

M. E. J. Newman and M. Girvan, Finding and evaluating community structure in networks, *Phys. Rev. E* 69, 026113 (2004).

A. Pocklington, M. Cumiskey, J. Armstrong and S. Grant, The proteomes of neurotransmitter receptor complexes form modular networks with distributed functionality underlying plasticity and behaviour. *Molecular System Biology* doi: 10.1038/msb4100041, (2006).

A. Pothen, H. Simon, and K.-P. Liou, Partitioning sparse matrices with eigenvectors of graphs. *SIAM J. Matrix Anal. Appl.* 11, 430-452 (1990).

F. Radicchi, C. Castellano, F. Cecconi, V. Loreto and D. Parisi, Defining and idetifying communities in networks. *Proc. Natl. Acad. Sci. USA* 101, 2658-2663 (2004).

S. Redner, How popular is your paper? An empirical study of the citation distribution. *Eur. Phys. J. B* 4, 131-134 (1998).

J. Reichardt and S. Bornholdt, Detecting Fuzzy Community Structures in Complex Networks with a Potts Model,*Phys. Rev. Lett.* 93, 218701 (2004).

J. Scott, *Social Network Analysis: A Handbook.* Sage, London, 2nd edition (2000).

S. H. Strogatz, Exploring complex networks. *Nature* 410, 268-276 (2001).

J. R. Tyler, D. M. Wilkinson, and B. A. Huberman, Email as spectroscopy: Automated discovery of community structure within organizations. In M. Huysman, E. Wenger, and V. Wulf (eds.), *Proceedings of the First International Conference on Communities and Technologies*, Kluwer, Dordrecht (2003).

S. Wasserman and K. Faust, *Social Network Analysis.* Cambridge University Press, Cambridge (1994).

D. J. Watts, *Small Worlds*. Princeton University Press, Princeton (1999).

D. J. Watts and S. H. Strogatz, Collective dynamics of 'small world' networks. *Nature* 393, 440-442 (1998).

R. J. Williams and N. D. Martinez, Simple rules yield complex food webs. *Nature* 404, 180-183 (2000).

F. Wu and B. A. Huberman, Finding communities in linear time: A physics approach. *Eur. Phys. J. B* 38, 331-338 (2004).

I. Xenarios, D. Eisenberg, Protein interaction databases. *Curr. Opin. Biotech.* 12, 334-339 (2001).

W. W. Zachary, An information flow model for conflict and fisson in small groups. *Journal of Anthropological Research* 33, 452-473 (1977).