Detecting network communities by constrained label propagation

M. J. Barber
ARC systems research

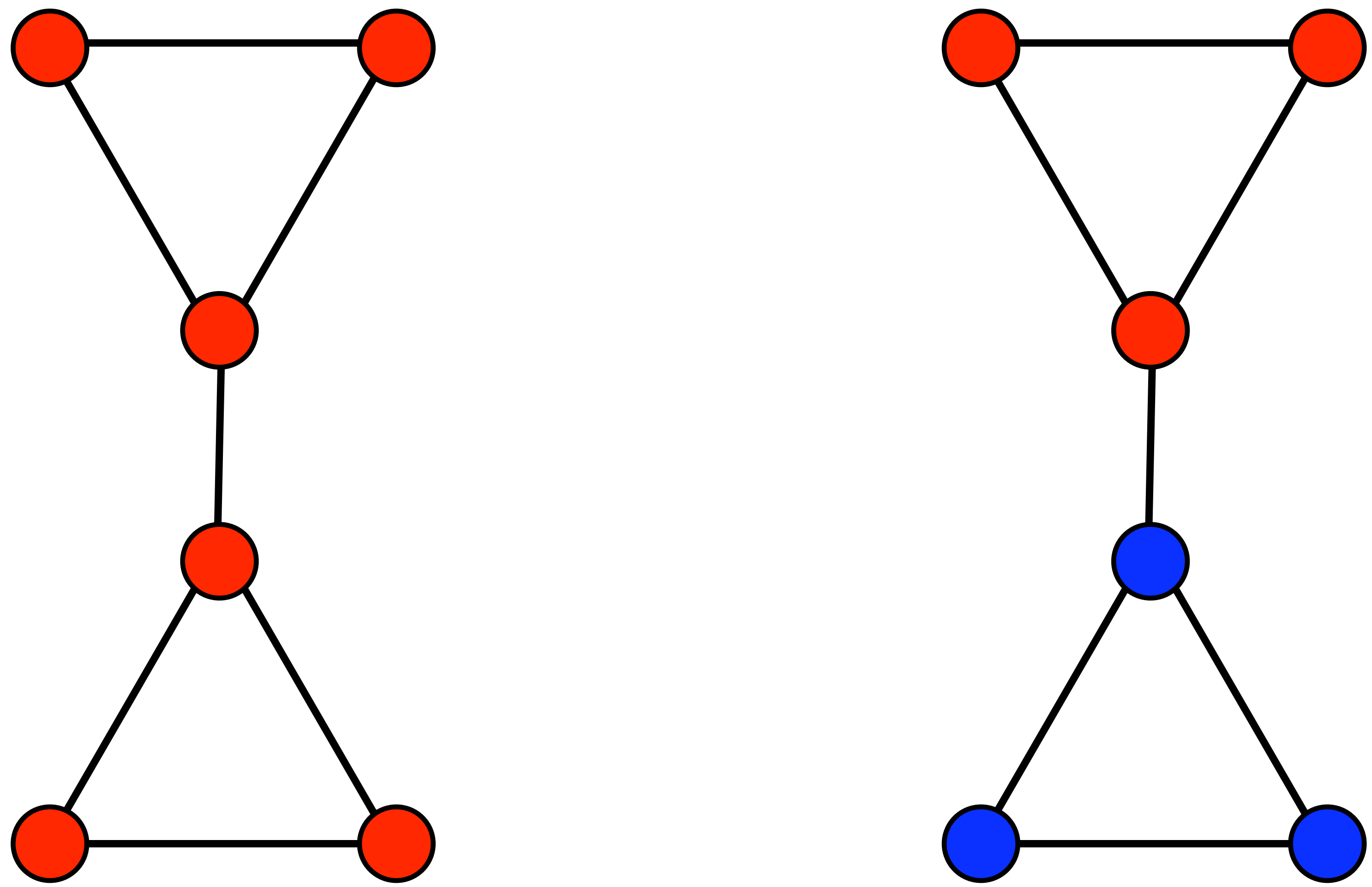
April 2, 2009
CCM, University of Madeira

Detecting network communities by propagating labels under constraints

Michael J. Barber and John W. Clark

http://arxiv.org/abs/0903.3138

Which communities?



Near linear time algorithm to detect community structures in large-scale networks

Usha Nandini Raghavan, Réka Albert, and Soundar Kumara

Phys. Rev. E 76(3):036106

Community detection and analysis is an important methodology for understanding the organization of various real-world networks and has applications in problems as diverse as consensus formation in social communities or the identification of functional modules in biochemical networks. Currently used algorithms that identify the community structures in large-scale real-world networks require a priori information such as the number and sizes of communities or are computationally expensive. In this paper we investigate a simple label propagation algorithm that uses the network structure alone as its guide and requires neither optimization of a pre-defined objective function nor prior information about the communities. In our algorithm every node is initialized with a unique label and at every step each node adopts the label that most of its neighbors currently have. In this iterative process densely connected groups of nodes form a consensus on a unique label to form communities. We validate the algorithm by applying it to networks whose community structures are known. We also demonstrate that the algorithm takes an almost linear time and hence it is computationally less expensive than what was possible so far.

Label Propagation Algorithm

1. Initialize: assign unique label to each vertex.
2. Relabel vertices by assigning most common label from the neighboring vertices.
   a. Keep the current label if it is among the most common
   b. Otherwise, pick a new label at random from the most common
3. Repeat relabeling until a stable set of labels has been found.

LPA Advantages

- Simple!
- Fast!
- Effective! (with quality measured by *modularity*)

LPA Disadvantages

- Community quality measure not clear from the algorithm.
- Allows poor solutions, e.g., all vertices in one community.
- Not actually that simple…

Label Propagation Mathematics

$$l'_v = \underset{l}{\operatorname{argmax}} \sum_{u=1}^{n} A_{uv} \delta\left(l_u, l\right)$$

$$H = \frac{1}{2} \sum_{v=1}^{n} \sum_{u=1}^{n} A_{uv} \delta\left(l_u, l_v\right)$$

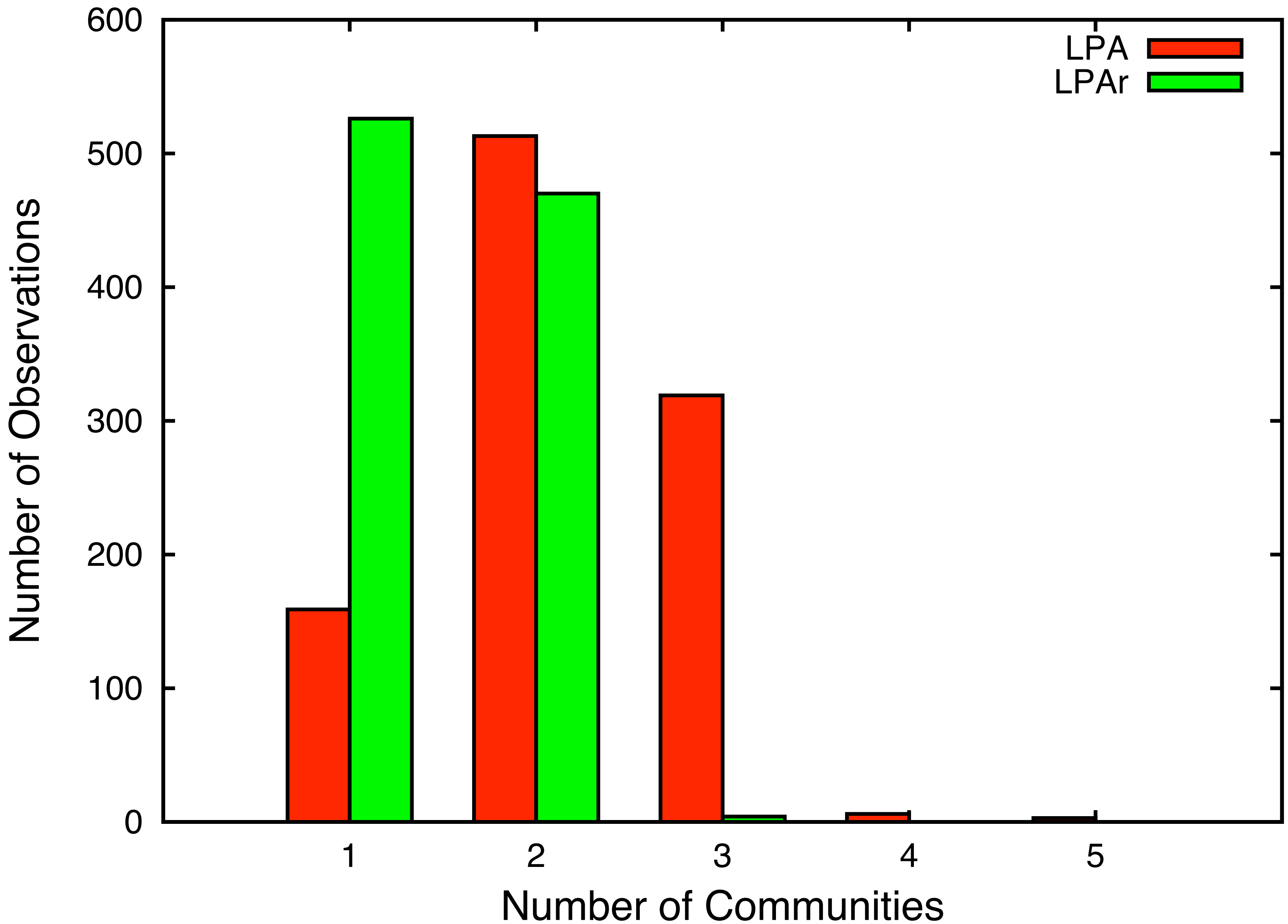
An Objective Function for the LPA

- LPA maximizes *H*, the number of edges that link vertices with the same label.
- Equivalent to minimizing the Hamiltonian for a ferromagnetic Potts model (Tibély and Kertész, 2008).
- Nothing prevents all vertices being assigned to the same community—it is the *globally optimal* solution!

Consequences

- Community solutions are local optima in the objective function.
- Community quality is not necessarily improved by the LPA.
- No way to compare communities (but can use auxiliary considerations such as the modularity).
- The LPA is hard to modify or optimize.

Dubious Optimization



LPA with Constraints

- Add a penalty term to the objective function.
- Can exclude undesirable solutions.
- Can give a reasonable measure of community quality.
- Can still be quite fast.

LPA for Modularity

H	number of edges linking vertices with the same label
D_i	number of edges between vertices with label i
G	$D_1^2 + D_2^2 + D_3^2 + \ldots + D_q^2$

$$H' = H - \lambda G$$

Performance

Maximum modularity (100 samples)

Network	LPA	LPAm	Hybrid
karate	0.4156	0.4000	0.4198
dolphins	0.5237	0.5157	0.5253
jazz	0.4424	0.4448	0.4442
network science	0.8924	0.8723	0.8934
condmat 2003	0.6228	0.5947	0.6360

Average modularity (100 samples)

Network	LPA	LPAm	Hybrid
karate	0.366(6)	0.347(3)	0.386(4)
dolphins	0.484(4)	0.4956(8)	0.495(3)
jazz	0.336(9)	0.4351(9)	0.366(7)
network science	0.8792(6)	0.8618(5)	0.8806(6)
condmat 2003	0.6073(6)	0.5828(4)	0.6139(9)

Conclusions

- LPA can be expressed as an optimization problem.
- Several hidden assumptions are built into the LPA.
- Constraints make it easier to understand how LPA works.

Detecting network communities by propagating labels under constraints

Michael J. Barber and John W. Clark

http://arxiv.org/abs/0903.3138