Università
della
Svizzera
italiana

Faculty
of Informatics

Institute of
Computational
Science
ICS

Platform for Advanced Scientific Computing

SoNAR-C Social Network Analysis Research Center

Università
della
Svizzera
italiana

InterDisciplinary
Institute of
Data Science
IDIDS

Efficient Markov Chain Monte Carlo Estimation of Exponential-Family Random Graph models

Maksym Byshkin[1], Alex Stivala[1,2], Antonietta Mira[1,4], Garry Robins[3], Alessandro Lomi[1,3]

[1]Università della Svizzera italiana, Lugano, Switzerland

[2]Swinburne University of Technology, Australia

[3]University of Melbourne, Australia

[4]Università dell'Insubria, Como, Italy

Second Australian Social Network Analysis Conference, Sydney, November 28-29, 2017

Introduction

We have developed efficient Markov chain-based algorithm to perform Maximum Likelihood parameter estimation for probability distributions with intractable normalizing constants

ERGMs are exponential family of probability distributions for dependent network data

$$\pi(x,\theta) = \frac{1}{k}\exp\left(\sum_A \theta_A z_A(x)\right)$$

$$k = \sum_x \exp\left(\sum_A \theta_A z_A(x)\right)$$

$z_A(x)$ are networks statistics (e.g. number of ties, triangles, stars ..)

Robins G, Snijders T, Wang P, Handcock M, & Pattison P (2007) Recent developments in exponential random graph (p*) models for social networks. *Social Networks* 29(2):192-215.

Introduction

We want to find MLE of the model parameters θ

$$E_{\pi(\theta)}\left(z_A(x)\right) = z_A(x_{obs})$$

$E_{\pi(\theta)}(z_A(x)) = \sum_x z_A(x)\pi(x,\theta)$ are expected statistics

Metropolis-Hastings algorithm may be used to compute $E_{\pi(\theta)}(z_A(x))$ and to generate **simulated data** $x(\theta)$

And we want to solve inverse problem: find $\theta(x_{obs})$

Lehmann, E.L., Casella, G. Theory of point estimation (2006)

Existing estimation approaches

1) Bayesian
2) Geyer-Thompson MCMCMLE
3) Method of Moments
   (Stochastic approximation)

- Iteratively update θ_A
- At many different θ_A values perform MCMC simulations to draw simulated data $x(\theta_A)$

⬇

MLE is computationally expensive

We can find MCMCMLE without MCMC simulation

Metropolis-Hastings algorithm

Given the state *x* the state *x'* is proposed with probability $q(x \rightarrow x')$

Acceptance probability $\alpha(x \rightarrow x', \theta_A) = \min\left\{1, \frac{q(x' \rightarrow x)\pi(x', \theta_A)}{q(x \rightarrow x')\pi(x, \theta_A)}\right\}$

Transition probability $P(x \rightarrow x', \theta_A) = q(x \rightarrow x')\alpha(x \rightarrow x', \theta_A)$

The algorithm generates a Markov chain x_t that converges to $\pi(x, \theta)$
if the algorithm step *t* is larger then the burn-in time

Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, & Teller E Equation of state calculations by fast computing machines. *The journal of chemical physics* 21, 1087-1092 (1953)
Hastings WK, Monte Carlo sampling methods using Markov chains and their applications. *Biometrika* 57, 97-109 (1970)

New approaches for MLE

I can show that for simulated networks drawn from $\pi(x,\theta)$ MLE may be found from Equilibrium Expectation (EE):

$$\mathrm{E}_{\pi(\theta)}\left(dz_A(x,\theta)\right) = 0$$

$$dz_A(x,\theta) = \sum_{x'} P(x \to x',\theta)\,(z_A(x') - z_A(x))$$

The left part may be computed by Monte Carlo integrations, without time consuming MCMC simulation

New approaches for MLE

If observed network x_{obs} is large then one sample is enough

$$\sum_{x'} P(x_{obs} \to x', \hat{\theta})\left(z_A(x') - z_A(x_{obs})\right) = 0$$

But may be used also for small networks

If we have data samples iid from $\pi(x, \theta)$: $x_{S_1}, x_{S_2}, .., x_{S_n}$

$$E_{\pi(\theta)}(dz_A(x,\theta)) = \frac{1}{n}\sum_i dz_A(x_{S_i}, \theta)$$

Very fast MLE !

MLE for empirical data

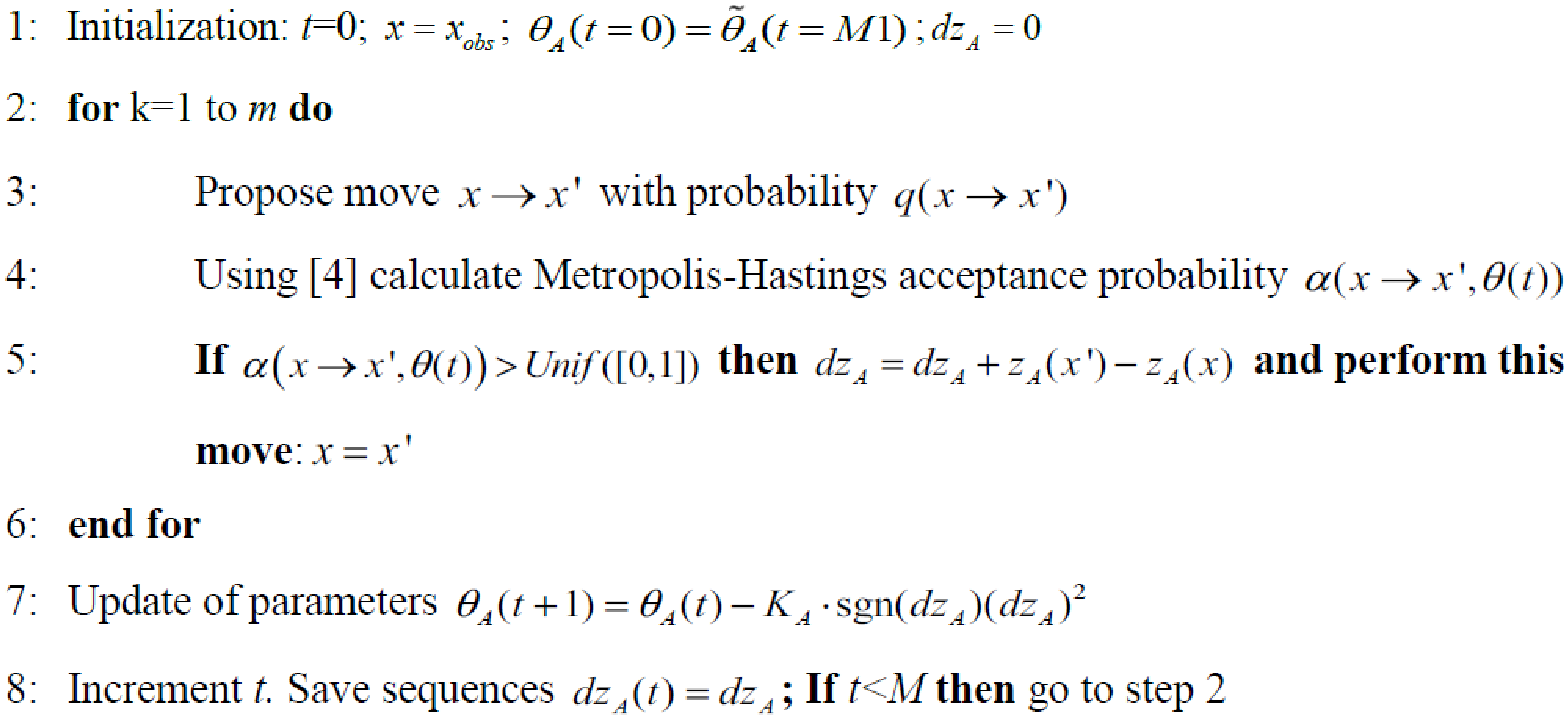
How to apply this methodology for empirical data x_{obs} ?
We can draw simulated data x so that $z_A(x) = z_A(x_{obs})$

$$\left.\begin{array}{l} \sum_{x'} P(x \to x', \theta_A)\left(z_A(x') - z_A(x)\right) = 0 \\ z_A(x) = z_A(x_{obs}) \end{array}\right\} \Rightarrow \quad E_\theta\left(z_A(x)\right) = z_A(x_{obs})$$

We have developed MCMC algorithm to solve this system of equation

EE algorithm for empirical data

MCMC to constrain the values of all the statistics



```
1: Initialization: $t$=0; $x = x_{obs}$; $\theta_A(t=0) = \tilde{\theta}_A(t=M1)$; $dz_A = 0$
2: for k=1 to $m$ do
3:         Propose move $x \to x'$ with probability $q(x \to x')$
4:         Using [4] calculate Metropolis-Hastings acceptance probability $\alpha(x \to x', \theta(t))$
5:         If $\alpha(x \to x', \theta(t)) > Unif([0,1])$ then $dz_A = dz_A + z_A(x') - z_A(x)$ and perform this
           move: $x = x'$
6: end for
7: Update of parameters $\theta_A(t+1) = \theta_A(t) - K_A \cdot \mathrm{sgn}(dz_A)(dz_A)^2$
8: Increment $t$. Save sequences $dz_A(t) = dz_A$; If $t$<$M$ then go to step 2
```

EE algorithm for empirical data

MoM (Snijders 2002) and MCMCMLE (Geyer CJ & Thompson 1992) require many converged outputs of Metropolis-Hastings algorithm

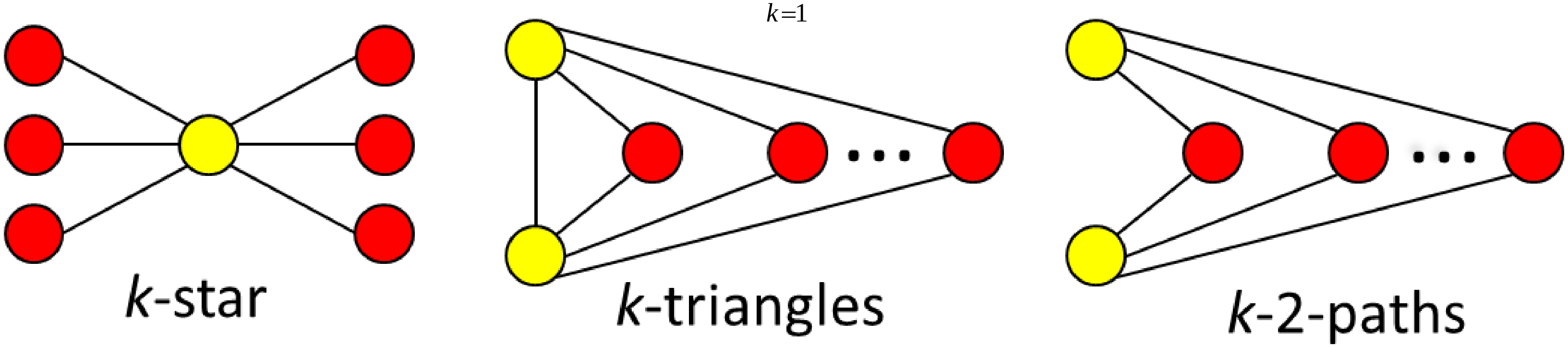
EE algorithm does not need such outputs. Instead it generates one converged output

EE algorithm is similar to Metropolis-Hastings algorithm, but allows Monte Carlo simulation to be performed while constraining the values of statistics $z_A(x)$ and in such a way that the EE condition is satisfied

Tests and applications

Networks statistics typical for social networks

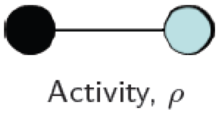
$$z_{AT}(x) = 3T_1(x) + \sum_{k=1}^{N-3} (-1)^k T_{k+1}(x) / \lambda^k$$



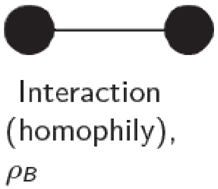
$$z_{AS}(x) = \sum_{k=2}^{N-1} (-1)^k S_k(x) / \lambda^{k-2}$$

$$z_{A2P}(x) = U_1(x) - \frac{2U_2(x)}{\lambda} + \sum_{k=3}^{N-2} (-1)^{k-1} U_k(x) / \lambda^{k-1}$$

Node attributes



$$z_\rho = \sum_{i,j} a_i x_{ij}$$



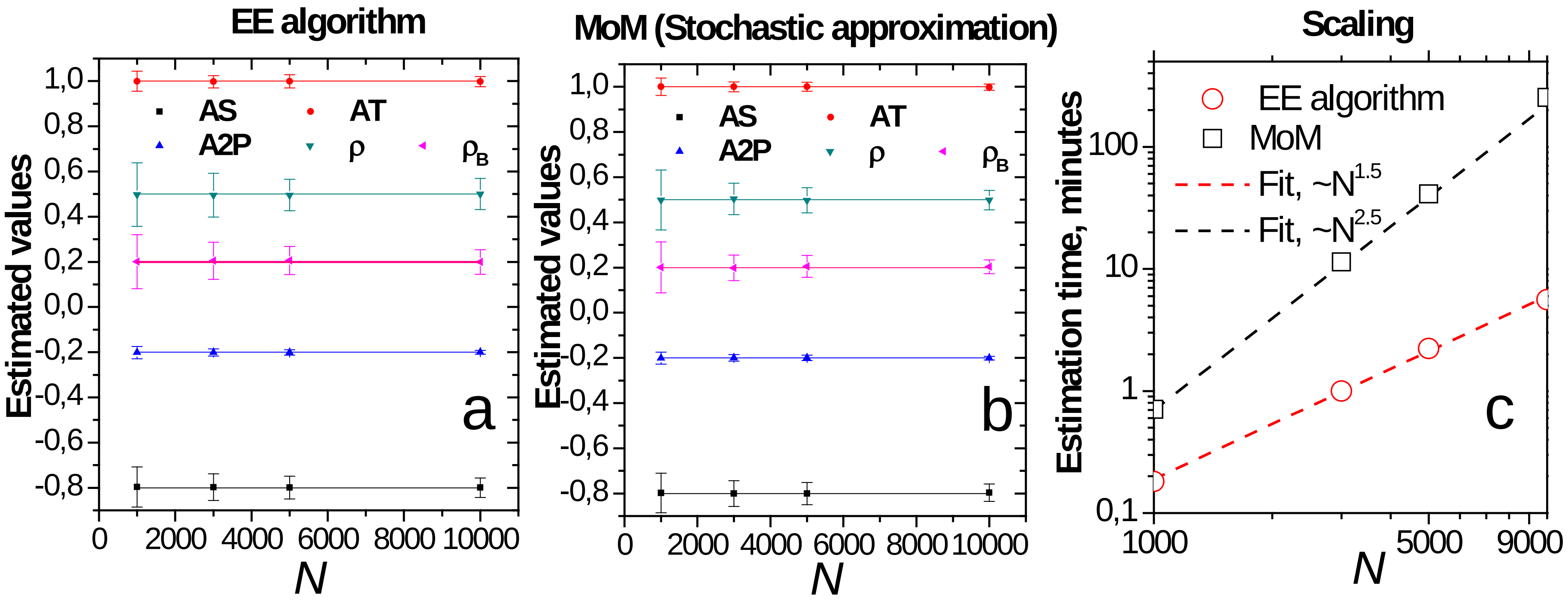
$$z_{\rho_B} = \sum_{i,j} x_{ij} a_i a_j$$

$$z_{Match} = \sum_{i,j} x_{ij} \delta_{a_i, a_j}$$

Snijders T.A., Pattison P.E., Robins G.L., Handcock M.S., New specifications for exponential random graph models, *Sociological Methodology* **36**(1), 99-153 (2006)

Tests and applications

We use EE algorithm to estimate parameters of ERGMs



With EE algorithm the estimation time grows with number of nodes *N* almost linear

Snijders, Tom AB. Markov chain Monte Carlo estimation of exponential random graph models, Journal of Social Structure, 1-40 (2002)

Tests and applications. Biological networks

Arabidopsis Interactome Mapping Consortium. Science, 333(6042):601–607, 2011.

C. von Mering, et al. Comparative assessment of large-scale data sets of protein–protein interactions. Nature, 417(6887):399–403, 2002

T. Rolland et al. A proteome-scale map of the human interactome network. Cell, 159(5):1212–1226, 2014.

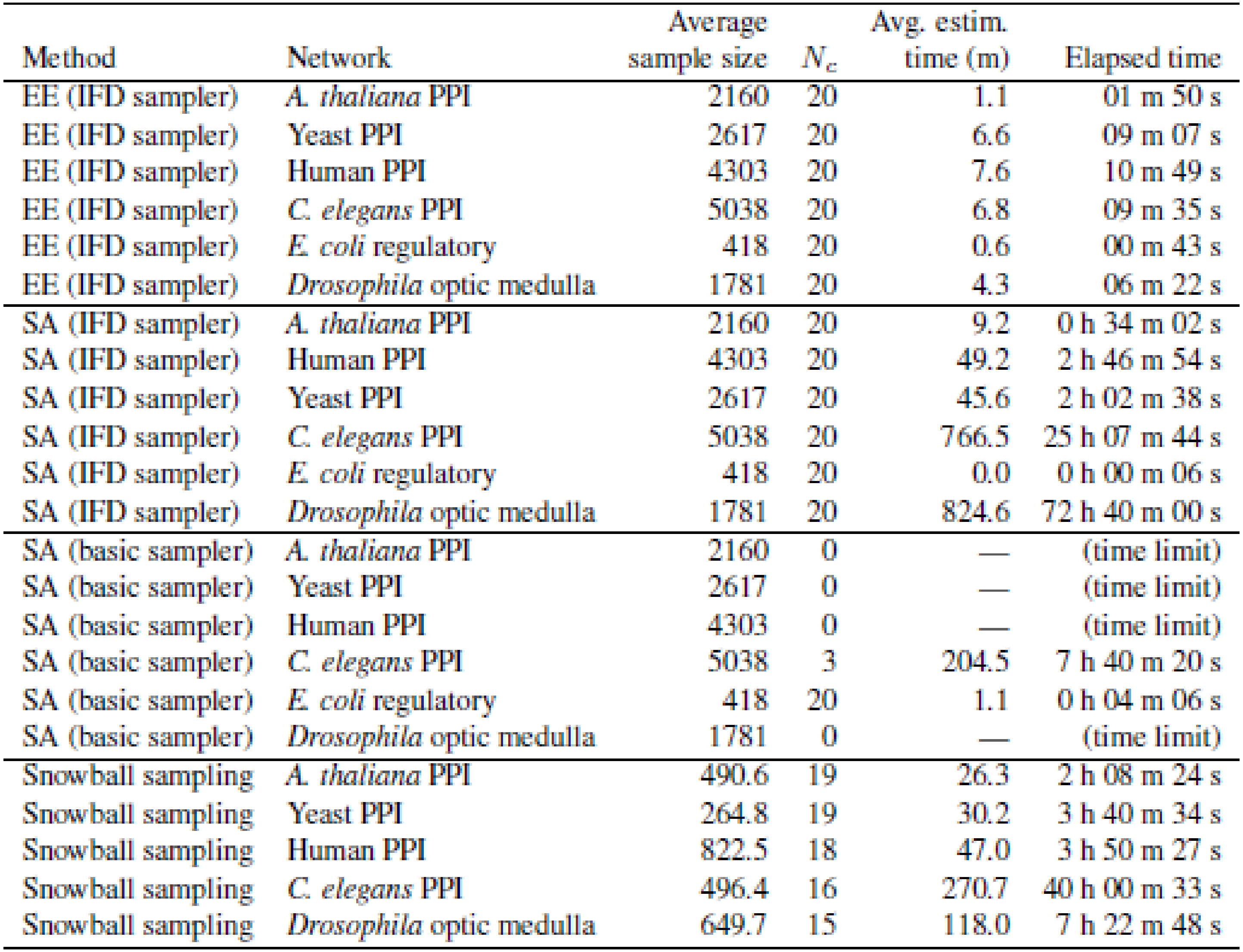
X.-T. Huang, et al An integrative C. elegans protein–protein interaction network with reliability assessment based on a probabilistic graphical model. Molecular BioSystems, 12(1):85–92, 2016.

S.-y. Takemura, et al. A visual motion detection circuit suggested by Drosophila connectomics. Nature, 500(7461):175–181, 2013

S. S. Shen-Orr et al, Network motifs in the transcriptional regulation network of Escherichia coli. Nature Genetics, 31(1):64–68, 2002.

Stivala, A. D., Koskinen, J. H., Rolls, D. A., Wang, P., & Robins, G. L. (2016). Snowball sampling for estimating exponential random graph models for large networks. *Social Networks*, *47*, 167-188.

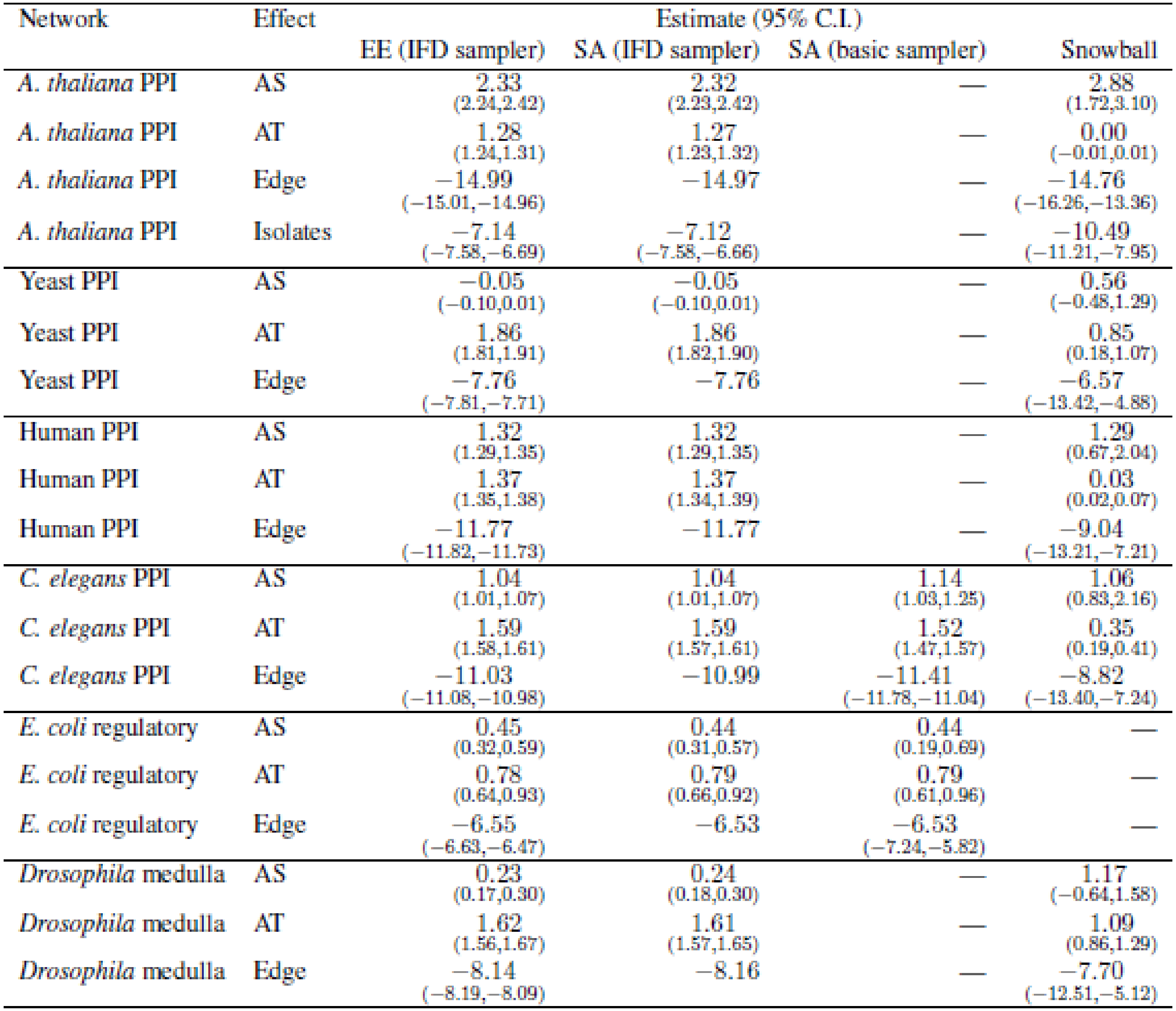
Tests and applications. Biological networks



Method	Network	Average sample size	N_c	Avg. estim. time (m)	Elapsed time
EE (IFD sampler)	*A. thaliana* PPI	2160	20	1.1	01 m 50 s
EE (IFD sampler)	Yeast PPI	2617	20	6.6	09 m 07 s
EE (IFD sampler)	Human PPI	4303	20	7.6	10 m 49 s
EE (IFD sampler)	*C. elegans* PPI	5038	20	6.8	09 m 35 s
EE (IFD sampler)	*E. coli* regulatory	418	20	0.6	00 m 43 s
EE (IFD sampler)	*Drosophila* optic medulla	1781	20	4.3	06 m 22 s
SA (IFD sampler)	*A. thaliana* PPI	2160	20	9.2	0 h 34 m 02 s
SA (IFD sampler)	Human PPI	4303	20	49.2	2 h 46 m 54 s
SA (IFD sampler)	Yeast PPI	2617	20	45.6	2 h 02 m 38 s
SA (IFD sampler)	*C. elegans* PPI	5038	20	766.5	25 h 07 m 44 s
SA (IFD sampler)	*E. coli* regulatory	418	20	0.0	0 h 00 m 06 s
SA (IFD sampler)	*Drosophila* optic medulla	1781	20	824.6	72 h 40 m 00 s
SA (basic sampler)	*A. thaliana* PPI	2160	0	—	(time limit)
SA (basic sampler)	Yeast PPI	2617	0	—	(time limit)
SA (basic sampler)	Human PPI	4303	0	—	(time limit)
SA (basic sampler)	*C. elegans* PPI	5038	3	204.5	7 h 40 m 20 s
SA (basic sampler)	*E. coli* regulatory	418	20	1.1	0 h 04 m 06 s
SA (basic sampler)	*Drosophila* optic medulla	1781	0	—	(time limit)
Snowball sampling	*A. thaliana* PPI	490.6	19	26.3	2 h 08 m 24 s
Snowball sampling	Yeast PPI	264.8	19	30.2	3 h 40 m 34 s
Snowball sampling	Human PPI	822.5	18	47.0	3 h 50 m 27 s
Snowball sampling	*C. elegans* PPI	496.4	16	270.7	40 h 00 m 33 s
Snowball sampling	*Drosophila* optic medulla	649.7	15	118.0	7 h 22 m 48 s

IFD sampler: Byshkin M, Stivala A, Mira A, Krause R, Robins G, Lomi A, Auxiliary Parameter MCMC for Exponential Random Graph Models J. Stat. Phys. 165: 740-754 (2016)

Tests and applications. Biological networks



Network	Effect	Estimate (95% C.I.)			
		EE (IFD sampler)	SA (IFD sampler)	SA (basic sampler)	Snowball
A. thaliana PPI	AS	2.33 (2.24,2.42)	2.32 (2.23,2.42)	—	2.88 (1.72,3.10)
A. thaliana PPI	AT	1.28 (1.24,1.31)	1.27 (1.23,1.32)	—	0.00 (−0.01,0.01)
A. thaliana PPI	Edge	−14.99 (−15.01,−14.96)	−14.97	—	−14.76 (−16.26,−13.36)
A. thaliana PPI	Isolates	−7.14 (−7.58,−6.69)	−7.12 (−7.58,−6.66)	—	−10.49 (−11.21,−7.95)
Yeast PPI	AS	−0.05 (−0.10,0.01)	−0.05 (−0.10,0.01)	—	0.56 (−0.48,1.29)
Yeast PPI	AT	1.86 (1.81,1.91)	1.86 (1.82,1.90)	—	0.85 (0.18,1.07)
Yeast PPI	Edge	−7.76 (−7.81,−7.71)	−7.76	—	−6.57 (−13.42,−4.88)
Human PPI	AS	1.32 (1.29,1.35)	1.32 (1.29,1.35)	—	1.29 (0.67,2.04)
Human PPI	AT	1.37 (1.35,1.38)	1.37 (1.34,1.39)	—	0.03 (0.02,0.07)
Human PPI	Edge	−11.77 (−11.82,−11.73)	−11.77	—	−9.04 (−13.21,−7.21)
C. elegans PPI	AS	1.04 (1.01,1.07)	1.04 (1.01,1.07)	1.14 (1.03,1.25)	1.06 (0.83,2.16)
C. elegans PPI	AT	1.59 (1.58,1.61)	1.59 (1.57,1.61)	1.52 (1.47,1.57)	0.35 (0.19,0.41)
C. elegans PPI	Edge	−11.03 (−11.08,−10.98)	−10.99	−11.41 (−11.78,−11.04)	−8.82 (−13.40,−7.24)
E. coli regulatory	AS	0.45 (0.32,0.59)	0.44 (0.31,0.57)	0.44 (0.19,0.69)	—
E. coli regulatory	AT	0.78 (0.64,0.93)	0.79 (0.66,0.92)	0.79 (0.61,0.96)	—
E. coli regulatory	Edge	−6.55 (−6.63,−6.47)	−6.53	−6.53 (−7.24,−5.82)	—
Drosophila medulla	AS	0.23 (0.17,0.30)	0.24 (0.18,0.30)	—	1.17 (−0.64,1.58)
Drosophila medulla	AT	1.62 (1.56,1.67)	1.61 (1.57,1.65)	—	1.09 (0.86,1.29)
Drosophila medulla	Edge	−8.14 (−8.19,−8.09)	−8.16	—	−7.70 (−12.51,−5.12)

This research was supported by Melbourne Bioinformatics at the University of Melbourne, grant number VR0261

Tests and applications. Large networks

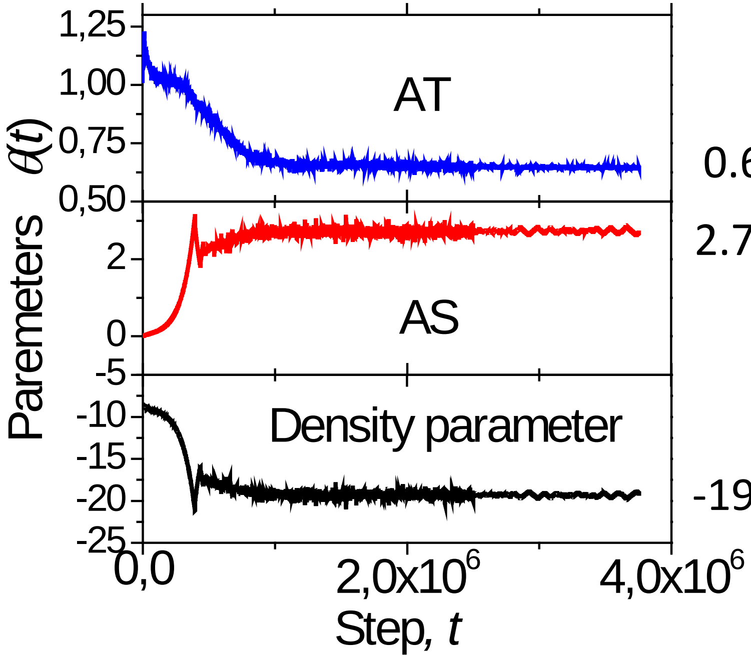
With good scaling we can obtain MLE for large dependent data

Livemocha was a social network for language learning

104103 nodes and 2193083 ties

Output of EE algorithm with IFD sampler (in 12 hours)

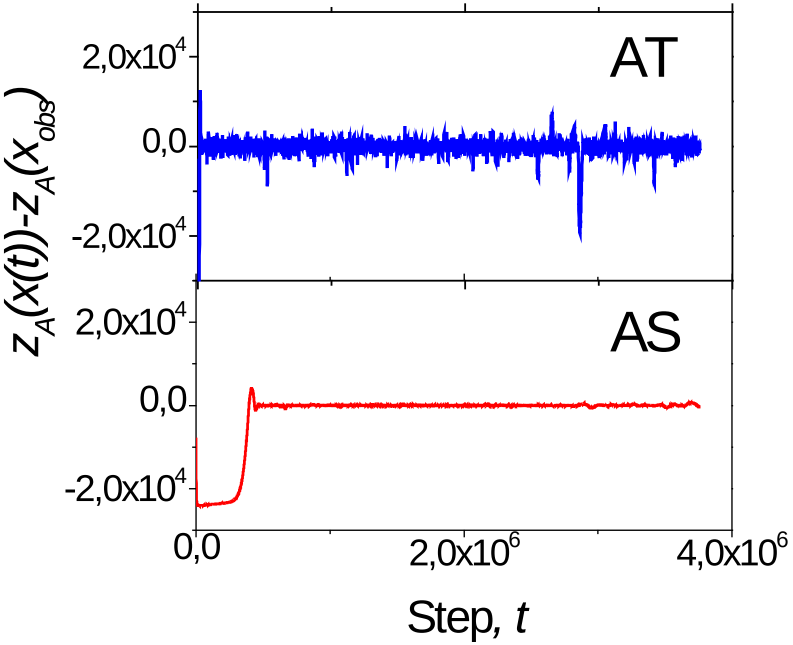
Statistics is constrained to the observed values



0.6242

2.7917

-19.531



Zafarani and H. Liu. Social computing data repository at ASU, 2009

Livmocha network datasets, konect.uni-koblenz.de

Tests and applications. Large networks

Livemocha

Networks with 104103 nodes and 2193083 ties

Both convergence tests are passed

Our convergence test

$$\left.\begin{aligned} &\theta_A(t)\ \text{converge} \\ &\left|\frac{\overline{z_A(x)-z_A(x_{obs})}}{SD(z_A(x)-z_A(x_{obs}))}\right|<0.1 \end{aligned}\right\}$$

⬇

$$\left.\begin{aligned} &\sum_{x'} P(x\to x',\theta_A)\big(z_A(x')-z_A(x)\big)=0 \\ &z_A(x)=z_A(x_{obs}) \end{aligned}\right\} \Rightarrow \quad E_{\pi(\theta)}\big(z_A(x)\big)=z_A(x_{obs})$$

t-ratio convergence test

$$t_A=\frac{E_{\pi(\theta)}\big(z_A(x)\big)-z_A(x_{obs})}{SD_{\pi(\theta)}\big(z_A(x)\big)}$$

$$|t_A|<0.3$$

⬇

Snijders T, Journal of Social Structure (2002): 1-40

More presentations

- Complex Networks, March 20-24, 2017, Dubrovnik, Croatia
- Sunbelt INSNA Conference, May 30-June 4, 2017, Beijing, China
- Cambridge Networks Day, 13th June 2017, Cambridge, UK
- PASC17, June 26 - 28, 2017, Lugano, Switzerland
- International Conference on Monte Carlo Methods and Applications, July 3-7, 2017, Montreal, Canada
- International Conference on Computational Social Science, July 10-13, 2017, Cologne, Germany
- Third European Conference on Social Networks, September 26 to 29, 2017, Mainz, Germany
- 2nd Australian Social Network Analysis Conference, 28-29 November 2017, Sydney, Australia

The code will be available free of charge

www.sonarcenter.eco.usi.ch

Hidden bonus slides

New approaches for MLE

The expected network statistics:

$$E_{\pi(\theta)}(z_A(x)) = \sum_x z_A(x)\pi(x,\theta)$$

The expectation must be taken with respect to $\pi(x,\theta)$

If on step *t* the chain is in state *x* what is the expected network statistics on the next step x′?

The expected network statistics:

$$\sum_{x'} z_A(x')P(x \to x',\theta)$$

The expectation must be taken with respect to $P(x \to x',\theta)$