Expanders, Tropical Semi-rings, and Nuclear Norms. Oh My!

Scientific computing for social and modern information networks.

By David Gleich

DOI: 10.1145/2425676.2425688

What does "The Matrix" have to do with "The Social Network"? Answering this question will take us on a tour of how scientific computing applies to many of the hottest problems studied in computer science today. From a grossly simplified point of view, scientific computing involves two intertwined activities: mathematical modeling of a problem and numerical computing in order to evaluate the model.

Consider the stereotypical textbook problem: How does heat diffuse through a plate? Physically, the problem corresponds with a linear partial differential equation (PDE). Numerically, a matrix problem or discrete time dynamical system then "solves" the PDE.

Scientific computing for social networks proceeds in the same way. Suppose a movie studio wishes to understand who might be interested in their latest sci-fi action thriller, and they have surveyed a small portion of people from Facebook to see if they like it. If your opinion of movies matches your friends' views on movies (links with this property often arise from a phenomena known as homophily in sociology [1]) then the answer to the movie studio's question corresponds to a social diffusion model. This diffusion gives rise to a numerical matrix problem that "solves" the model (see Figure 1). This problem is called semi-supervised learning on networks [2], and it applies whenever we have a partially labeled network with homophily links. More importantly for this article, it represents one relationship between matrices and social networks.

Matrices that arise from social networks can, in many senses, be incredibly different from the matrix problems that arise in scientific computing for physical sciences. One key difference is that social networks fall into a group known as "expanders"; if you look at the friends of any small group of people, it usually expands to another group of roughly the same size. For connected discrete sets like networks, this property is often explained as a surface-area-to-volume ratio. The volume is the size of the set, and the surface area is the size of the boundary set of connected elements. In a social network, the volume is the number of people inside a group and the surface area is the number of friends connected to the group, but not inside. The fact that these two sets have roughly the same size is radically different from the surface area to volume ratio of any physical problem. Once we discretize a physical space, just think about how the number of points inside of a spherical shape grows in relationships to the number of points on the boundary!

(See Figure 2 for an illustration of the effect.) As a result, expander graphs yield matrix problems with incredibly intricate connections that challenge off-the-shelf parallel matrix libraries [3]. Dealing with this problem is only one of the challenges posed by scientific computing for social networks.

A bigger challenge is that checking solutions is non-trivial. For the problem of heat diffusion in a plate, it isn't too difficult to setup a small experiment in order to check the answer from your simulation. Usually, we understand enough about the physics to spot "non-physical" PDE solutions instantly, but we have no such intuition about the physics of social networks. So to check an answer to the "movie diffusion" question posed earlier, you'd have to survey individuals to determine whether they'd seen the movie, and given that, if they liked it. Such experiments are often done on a small scale, but imagine trying to check your answers for the social network of a large university with 30,000 students. Surveying even a small portion of them would be arduous. Thus, checking if your answer is "non-physical" is often impossible for social network research. There are three common alternatives.

The first is to cook up a synthetic problem where you know the answer and check that you get it. The second is to evaluate the predictive power of a model on a standard test data set—usually collected and distributed by another researcher—using a cross-validation approach. Researchers have found social prediction tasks benefit by combining many different types of information; thus a third, weaker, form of validation is sometimes appropriate: Demonstrate that adding your new solution to an ensemble of existing solutions enables better predictions for some tricky problem.

Matrices that arise from social networks can, in many senses, be incredibly different from the matrix problems that arise in scientific computing for physical sciences.

The PDE models in physical problems are typically derived from accepted physical laws. We do not yet have any completely accepted "social network laws." Without such laws, we run into two problems. First, all too often, I see research that applies a state of the art method from scientific computing to a social dataset. Interesting but anecdotal results are shown, and the method is declared successful. Without any laws, it's difficult to argue. But, because there was no model guiding the particular computations, it's hard to judge how these methods will do in the future. Instead, researchers should first define and model the problem they wish to solve, and then determine how this problem gives rise to an algorithmic method.

For instance, there are two types of social diffusion. The first models a diffusion of a conserved quantity (think of attention or importance, which are fun-

Figure 1. In a standard scientific computing problem, we find the steady state heat distribution of a plate with a heat-source in the middle. This scientific problem is solved via a linear system. In a social diffusion problem, we are trying to find people who like the movie (labeled in dark orange) instead of people who don't like the movie (labeled in dark purple). By solving a different linear system, we can determine who is likely to enjoy the movie (light orange).

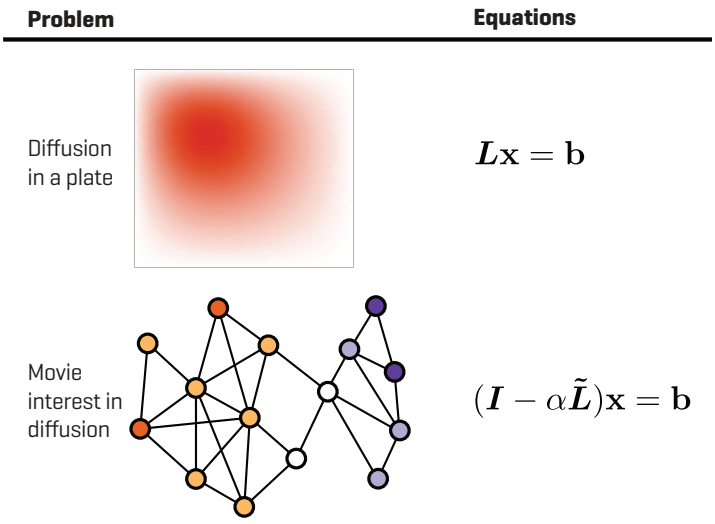
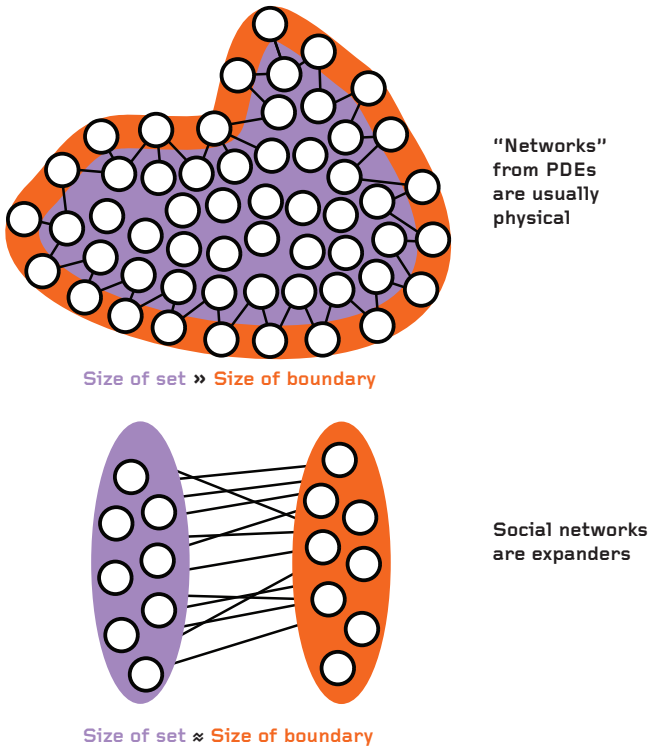


Figure 2. The network, or mesh, from a typical problem in scientific computing resides in a low dimensional space—think of two or three dimensions. These physical spaces put limits on the size of the boundary or "surface area" of the space given its volume. No such limits exist in social networks and these two sets are usually about the same size. A network with this property is called an expander network.



damentally limited); the second type models a diffusion of a virtual good (think of sending links or a virus spreading on a network), which can be copied infinitely often. Google's celebrated PageRank model uses a conservative diffusion to determine importance of pages on the Web [4], whereas those studying when a virus will continue to propagate found the eigenvalues of the non-conservative diffusion determine the answer [5]. Thus, just as in scientific computing, marrying the method to the model is key for the best scientific computing on social networks.

Ultimately, none of these steps differ from the practice of physical scientific computing. The challenges in creating models, devising algorithms, validating results, and comparing models just take on different challenges when the problems come from social data instead of physical models. Thus, let us return to our starting question: What does the matrix have to do with the social network? Just as in scientific computing, many interesting problems, models, and methods for social networks boil down to matrix computations. Yet, as in the expander example above, the types of matrix questions change dramatically in order to fit social network models. Let's see what's been done that's enticingly and refreshingly different from the types of matrix computations encountered in physical scientific computing.

EXPANDER GRAPHS AND PARALLEL COMPUTING

Recently, a coalition of folks from academia, national labs, and industry set out to tackle the problems in parallel computing and expander graphs. They established the Graph 500 benchmark (http://www.graph500.org) to measure the performance of a parallel computer on a standard graph computation with an expander graph. Over the past three years, they've seen performance grow by more than 1,000-times through a combination of novel software algorithms and higher performance parallel computers. But, there is still work left in adapting the software innovations for parallel computing back to matrix computations for social networks.

MATRIX COMPUTATIONS WITH SEMI-RINGS

One of the neatest uses for matrix computations on social networks is to compute standard graph algorithms, such as breadth-first searches, connected components, shortest paths, minimum spanning trees, and matchings, by restating the graph algorithm as a matrix computation over a semi-ring. A semi-ring is a more general number system than the real and complex numbers. Practically speaking, using a semi-ring approach often involves something like a matrix-vector multiplication where the meanings of addition, multiplication, 1, and 0 have all changed.

For instance, in computing a shortest path, "a × b" in a matrix equation is reinterpreted as "add a and b" and "a + b" is reinterpreted as "take the minimum of a and b." The resulting algebraic structure is called the "tropical semi-ring." Using it along with standard parallel matrix constructions enables highly scalable computations for social networks. See, for instance, the combinatorial BLAS package [6]. Many of the MapReduce-based graph computations also use these formulations [7]. For more information, see the excellent edited volume, *Graph Algorithms in the Language of Linear Algebra* [8].

SPARSE, LOCAL MATRIX COMPUTATIONS

How can you solve a linear system without even looking at all the elements of the matrix? In general, you can't but for a class of problems that exhibit a property called "localization" you can find a good approximation of the solution. Many of the linear systems that arise in social network analysis have these properties and we can compute localized solutions without even looking at the entire matrix [9]. The algorithm does a greedy-like exploration of the network around the values where the right hand side is large. For one particular type of social diffusion, we were able to find a good approximate solution on a network with 100,000,000 links in under a second [10]. When you can exploit this type of structure in a problem, it means that you don't need a large computer to solve it, but a laptop instead.

Most of the techniques above apply to almost any problem in matrix computations for social networks. To get more insight out of networks of data, though, researchers are also proposing new types of matrix models customized for social networks.

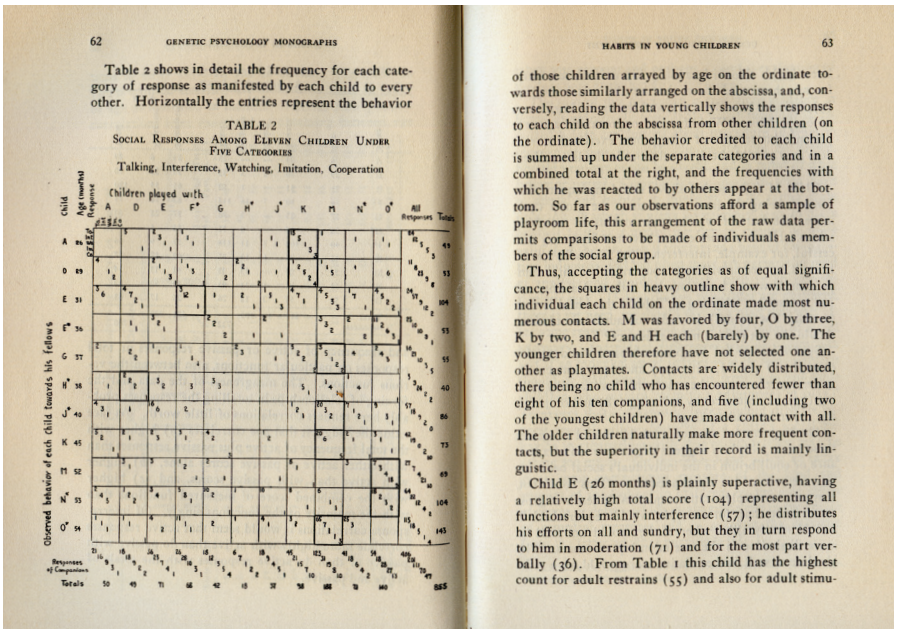
FUNCTIONS OF MATRICES

Think back to your ordinary differential equations (ODEs) class. How did you solve a 2x2 system of linear ODEs? You probably remember exponentiating the system represented as a matrix, but this didn't mean computing the exponential of each element, instead you looked at a function of a matrix. The same methodology applies to social network problems such as determining the importance of a node in a network. In fact, the matrix exponential of the adjacency matrix of a network computes a particular type of social diffusion. More complex models include using hyperbolic sines and cosines. But, evaluating functions of matrices with large networks required a new type of matrix computation. Golub and Meurant provided the basis for these innovations in their groundbreaking work on matrices, moments, and quadrature [11]. These methods have now been utilized and extended for the problems arising in network research [10, 12, 13]. This is the best type of research in the area, when new models give rise to new applications of algorithmic innovations.

UNCERTAINTY QUANTIFICATION

Virtually all computational models have some parameters. In the heat-diffusion-in-a-plate, how conductive is the plate? In the movie-social-diffusion, how likely is movie interest to follow a link? In order to understand the effect of variability in these parameters, the uncertainty quantification (UQ) community proposed a large set of new models and techniques for physical scientific computations. These same models and techniques also apply to social and information networks [14], where they can yield new insights about the importance of nodes in a network or about the sensitivity of these importance scores. In my past work on this topic, I used a method from UQ in order to create a new type of sensitivity measure for Web pages. It helped us to improve a detection algorithm for whether or not a Web page is spam and led us to look

Figure 3. A social network of children's interactions in five categories treated with a 3-D matrix-like data structure from Bott. This figure is one of the earliest applications of matrix and tensor methods to social networks. The image is taken from H. Bott's "Observation of Play Activies in a Nursery School" published in *Child Behavior, Differential and Genetic Psychology*.



62 GENETIC PSYCHOLOGY MONOGRAPHS

Table 2 shows in detail the frequency for each category of response as manifested by each child to every other. Horizontally the entries represent the behavior

TABLE 2

SOCIAL RESPONSES AMONG ELEVEN CHILDREN UNDER FIVE CATEGORIES

Talking, Interference, Watching, Imitation, Cooperation

HABITS IN YOUNG CHILDREN 63

of those children arrayed by age on the ordinate towards those similarly arranged on the abscissa, and, conversely, reading the data vertically shows the responses to each child on the abscissa from other children (on the ordinate). The behavior credited to each child is summed up under the separate categories and in a combined total at the right, and the frequencies with which he was reacted to by others appear at the bottom. So far as our observations afford a sample of playroom life, this arrangement of the raw data permits comparisons to be made of individuals as members of the social group.

Thus, accepting the categories as of equal significance, the squares in heavy outline show with which individual each child on the ordinate made most numerous contacts. M was favored by four, O by three, K by two, and E and H each (barely) by one. The younger children therefore have not selected one another as playmates. Contacts are widely distributed, there being no child who has encountered fewer than eight of his ten companions, and five (including two of the youngest children) have made contact with all. The older children naturally make more frequent contacts, but the superiority in their record is mainly linguistic.

Child E (26 months) is plainly superactive, having a relatively high total score (104) representing all functions but mainly interference (57); he distributes his efforts on all and sundry, but they in turn respond to him in moderation (71) and for the most part verbally (36). From Table 1 this child has the highest count for adult restrains (55) and also for adult stimu-

Child Behavior, Differential and Genetic Psychology. Clark University, 1928, 44–88.

into measuring how often people actually click links on the Web [15].

TENSOR METHODS

Why do we have to work with two-dimensions matrices? We don't! The study of tensor methods for data arrays with three or more dimensions has exploded over the past 10 years. Kolda and Bader did a fantastic job summarizing much of the recent work in the area [16]. What is perhaps surprising is that tensor methods for social networks are actually very old. Some of the methods date back to psychology literature in the 1920s and 1930s (see Figure 3 for an example). Recent work has shown many tensor problems are much harder than their matrix analogs—usually NP-hard [17]. Nevertheless, these new models are frequently used to derive insight into social networks, for example Dunlavy et al. studied temporal patterns [18].

MODERN MATRIX NORMS

Given a matrix, how large or complex is it? Matrix norms answer this question. In most matrix computations classes, it is my experience that matrix norms beyond the common Frobenius and p-norms are rarely covered. (Obviously, some exceptions apply.) But there is an enormous diversity of matrix norms beyond these textbook cases. For instance, in order to predict movie ratings in the Netflix dataset, Candes et al. wanted to find the simplest matrix that fit the data [19]. They used the idea of matrix rank as a measure of simplicity. Unfortunately, solving this problem exactly, in general, is intractable in the NP-hard sense of the word. However, they were able to solve an important special case by using the nuclear norm of a matrix as a proxy for rank. In machine learning on networks, max-norms are used to measure a different type of complexity and cut-norms are a different type of measure. Even evaluating the cut-norm for a general matrix is an NP-hard problem. These modern norms enrich our ability to model problems on social networks, and they are fruitful areas for future research due to their recency.

And so, what does the future hold? Hopefully more of the same. The best research in the future will follow the same patterns: find a problem on a social network, determine a realistic model, and then decide on a computable method to solve the model. Hopefully, I've convinced you of the usefulness of stating problems as matrix problems. To continue to do this in the future, the field needs interdisciplinary students and researchers who are educated in probability, machine learning, large-scale matrix computations, parallel computing, scientific computing, and statistics. To conclude, here are two problems I think are due for breakthroughs soon.

Usually, we understand enough about the physics to spot "non-physical" PDE solutions instantly, but we have no such intuition about the physics of social networks.

Model Reduction. We don't solve physical problems with individual atoms and particles of matter. Instead, we've built laws in order to abstract or average these details away. Yet, all of the computations discussed work on each-and-every node and edge in graphs with over a billion nodes. We need fresh ideas on how to leverage these models or abstract the details.

Higher Order Analysis. I also think there will be many interesting problems and research that arises out of thinking of social networks as collections of higher order groups. Finite elements, for instance, are usually defined as polynomials on the face of a triangle or other geometric structure. Why not look at social networks as collections of triangles where all three individual edges exist? New tensor methods will be key components of these analyses.

References

[1] McPherson, M., Smith-Lovin, L. and Cook, J. M. Birds of a Feather: Homophily in social networks. *Annual Review of Sociology, Annual Reviews*, 27,1 (2001), 415-444.

[2] Zhou, D., Bousquet, O., Lal, T. N., Weston, J., and Schölkopf, B. Learning with Local and Global Consistency. *NIPS*, 2003.

[3] Gleich, D. F. and Zhukov, L. Scalable Computing with Power-Law Graphs: Experience with Parallel PageRank. Poster. SuperComputing 2005. November 2005.

[4] Page, L., Brin, S., Motwani, R. and Winograd, T. The PageRank Citation Ranking: Bringing Order to the Web. Stanford University, 1999.

[5] Wang, Y.,Chakrabarti, D., Wang, C., and Faloutsos, C. Epidemic spreading in real networks: An eigenvalue viewpoint. In the *Proceedings of the 22nd International Symposium on Reliable Distributed Systems* (Florence, Italy, Oct. 6-8). IEEE, Washington, D.C., 2003, 25–34.

[6] Buluç, A and Gilbert, J. R. The combinatorial BLAS: Design, implementation, and applications. *International Journal of High Performance Computing Applications (IJHPCA)* 25, 4 (2011), 496-509.

[7] Kang, U., Tsourakakis, C., and Faloutsos, C. PEGASUS: A peta-scale graph mining system implementation and observations. In the *ICDM '09 Proceedings of the Ninth IEEE International Conference on Data Mining* (Miami, FL, Dec. 6-9), IEEE, Washington, D.C., 2009, 229 -238.

[8] Kepner, J. and Gilbert, J. (Eds.) *Graph Algorithms in the Language of Linear Algebra*. SIAM, Philadelphia, PA, 2011.

[9] Leskovec, J., Lang, K. J., Dasgupta, A., and Mahoney, M. W. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics* 6,1 (2009), 29-123.

[10] Bonchi, F., Esfandiar, P., Gleich, D. F., Greif, C., and Lakshmanan, L. V. Fast matrix computations for pairwise and columnwise commute times and Katz scores. *Internet Mathematics* 8, 1-2 (2012), 73-112.

[11] Golub, G. and Meurant, G. Matrices, moments and quadrature II: How to compute the norm of the error in iterative methods. *BIT Numerical Mathematic*, 37, 3 (1997), 687-705.

[12] Benzi, M. and Boito, P. Quadrature rule-based bounds for functions of adjacency matrices. *Linear Algebra and its Applications*, 433, 3 (2010), 637-652.

[13] Estrada, E. and Higham, D. J. Network properties revealed through matrix functions. *SIAM Review* 52, 4 (2010), 696-714.

[14] Constantine, P. G. and Gleich, D. F. Random alpha PageRank. *Internet Mathematics*, 6, 2 (2010), 189-236.

[15] Gleich, D. F., Constantine, P. G., Flaxman, A. and Gunawardana, A. Tracking the random surfer: empirically measured teleportation parameters in PageRank. In the *Proceedings of the 19th international conference on World Wide Web* (Raleigh, NC, April 26-30). ACM, New York, 2010, 381-390.

[16] Kolda, T. G. and Bader, B. W. Tensor decompositions and applications. *SIAM Review* 51, 3 (2009), 455-500.

[17] Hillar, C. and Lim, L.-H. Most tensor problems are NP-hard. *arXiv*, 2009, cs.NA, 0911.1393.

[18] Dunlavy, D. M., Kolda, T. G., and Acar, E. Temporal link prediction using matrix and tensor factorizations. *ACM Transactions on Knowledge Discovery from Data* 5, 2 (2011), 10:1-10:27.

[19] Candès, E. J. and Recht, B. Exact matrix completion via convex optimization. *Foundations of Computational Mathematics* 9, 6 (2009), 717-772.

Biography

David F. Gleich is an assistant professor of computer science at Purdue University, where he probes the parallels between network problems and matrix methods. He holds a joint B.S. degree in computer science and mathematics from Harvey Mudd College, and M.S. and Ph.D. degrees from Stanford University in computational and mathematical engineering.