

Propagation and Immunization in Large Networks

Many interesting research questions can be explored by studying processes running over networks.



By *B. Aditya Prakash*

DOI: 10.1145/2331042.2331059

How do contagions spread in populations? Who are the best people to vaccinate? Which group should we market to for maximizing product penetration? Will a given YouTube video, meme, or link go viral? And what happens when two products compete?

One feature these questions have in common is they are all important research problems (see the titles listed at the end of this article for further reading). The other is that they all can be characterized as problems based on giant graphs (networks) of nodes (people) and edges (relationships). Such networks are ubiquitous, from online social networks and

gene-regulatory networks to router graphs. Networks effectively model a wide range of phenomena by exposing local-dependencies while simultaneously capturing large-scale structure. Questions such as how blackouts can spread on a nationwide scale, how social systems evolve on the basis of individual interactions, or how efficiently we can search data on large networks of blogs or websites, are all related to phenomena on networks. Clearly progress here holds great scientific as well as commercial value.

Big data is a natural and necessary part of research in this sphere. Although the actions of a particular individual or component may be too difficult to model, data mining and machine learning can be applied to large groups or ensembles, in turn yielding effective models with the

ability to predict future events. For instance, modeling the response of every individual to a particular marketing strategy might be too difficult, but modeling the behavior of large groups of people based on demographics and geography is feasible. Models are useful as they allow us to abstract out the process and simulate it on our machines, and we can then try to explore even more complex issues using these models. For example, how should we distribute resources to control an epidemic? How should we manage communities to make them more productive? And how can we design these policies so that they can be implemented on an extremely large-scale?

Invariably, solving such problems involves working with huge amounts of data—millions of users, billions of tweets, and trillions of network con-

nections—as well as designing algorithms and experiments using generated models, which can themselves be run on large and complex data. Two trends have emerged to allow such an approach: The increasing ability to collect more and more data, and the increasing ability to run more and more large-scale and complex models. In the past, when the amount of data available was small and computing power was limited, researchers used markedly different approaches. Sociologists, for example, used to collect small samples and then extrapolate to develop very sophisticated models. We are now in a position to do the opposite. Through the emergence of big data, we can develop and test increasingly abstracted models on larger and larger sample sizes.

Dynamical processes over net-

works can give rise to astonishing macroscopic behavior, leading to challenging and exciting research problems. How stable is a predator-prey ecosystem, given intricate food webs? How do rumors spread on Twitter/Facebook? How should we administer software patches optimally? Herein, we will try to illustrate some big-data challenges using two problems related to dynamical phenomena (like propagation) on large networks: thresholds and immunization. Their applications are broad, and these problems are central to surprisingly diverse areas including cyber security, epidemiology, and public health, through to product marketing and information dissemination.

TIPPING POINTS AND THRESHOLDS

Consider the following problem: Given a network of who-contacts-whom, will a contagious virus “take-over” (cause an epidemic) or die-out quickly? What will change if nodes have partial, temporary, or permanent immunity? What if the underlying network changes over time, e.g., if people have different connections during the day at work, and during the night at home? An important concept in answering these questions is the “epidemic threshold,” which is the minimum level of virulence required to prevent a viral contagion from dying out quickly. Determining the epidemic threshold is a fundamental question in epidemiology and related areas.

Apart from the fundamental nature of this problem, it turns out that it is also very helpful in running large-scale epidemiological simulations. While it is very difficult to model each and every person’s response to a disease, it is much more feasible to run epidemic simulations on huge populations (city or nationwide) to understand which sections get infected, which should be quarantined, and so on. One problem here is that running big simulations—potentially involving hundreds of machines—is very expensive. How can we speed up these simulations to enable more useful and more frequent runs? The epidemic threshold problem we described above comes to our need: We don’t need to run simulations when the disease or virus in question

is “below-threshold,” thus speeding-up the simulations.

Surprisingly, it can be shown that when the underlying contact-network does not change over time [1], the threshold condition is,

$$\lambda_1 C < 1$$

Where λ_1 is the first eigenvalue of the connectivity matrix, and C is a virus-model dependent constant. This holds true for (a) any graph; and (b) all propagation models in standard literature, including the AIDS virus HIV and more than 25 others from canonical texts [2]. So, the result we achieve decouples the effect of the topology and the virus model. What makes the result practical is the eigenvalue computation on graphs is linear-time in

the size of the graph, and also can be efficiently parallelized on Hadoop [3].

What exactly is λ_1 ? Algebraically, it is simply the eigenvalue of the underlying adjacency matrix with the largest magnitude. Intuitively though, it captures how vulnerable the graph is for an epidemic (a concept which will prove useful later too). Roughly, it describes the number of paths between pairs of nodes in a graph, discounting for longer paths, effectively controlling the number of ways the virus can spread. Hence, the larger is λ_1 , the better the graph’s connectivity for the virus (see Figure 1).

Figure 2 demonstrates the result of computer simulation experiments on a large public dataset representing a synthetic population of the city of Portland, OR [4]. The dataset is based

Figure 1. Why λ_1 matters more than number of edges E . Changing connectivity and vulnerability of graphs with changing λ_1 . The clique [largest λ_1] is the most vulnerable. Note that E is not enough. Star and chain have the same number of edges [$E = 4$], but the star is intuitively more vulnerable [it also has a higher λ_1].

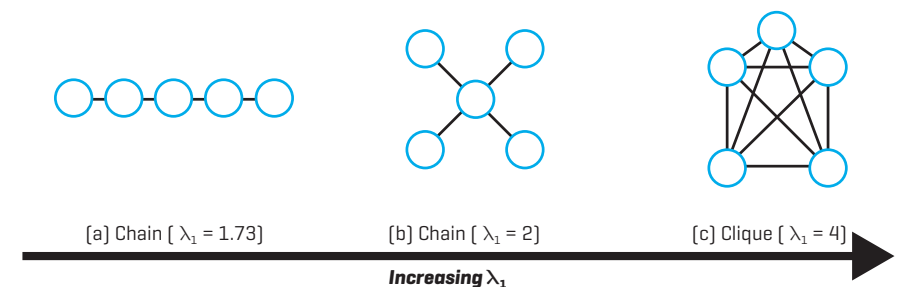
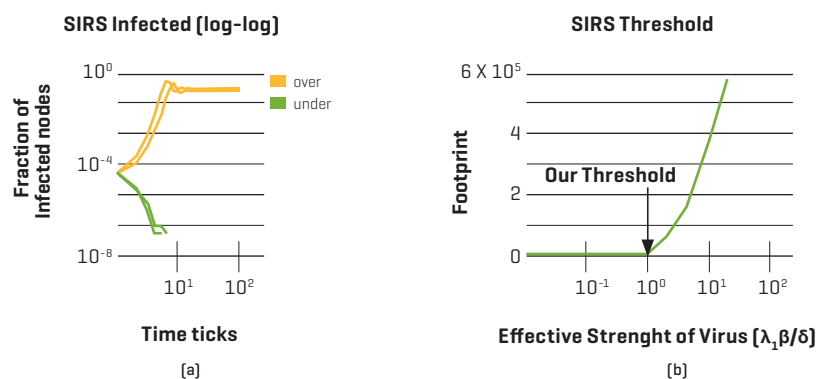


Figure 2. The tipping-point. Simulation results on a massive social-contact graph PORTLAND [31 mil. edges, 1.5 mil. nodes] and the SIRS model [temporary immunity like pertussis]. (a) Plot of Infected Fraction of Population versus Time [log-log]. Note the qualitative difference in behavior under (green) the threshold and above (red) the threshold. (b) Footprint [expected final epidemic size] versus Effective Strength [lin-log]. Notice the prediction is exactly at the take-off point.



on detailed microscopic simulation-based modeling and integration techniques, and has been used in modeling studies on smallpox outbreaks. It is a social-contact graph containing 31,204,286 links (interactions) among 1,588,212 nodes (people). The simulations were conducted using the so-called “SIRS” model, which models diseases to which we can become temporarily immune (like *pertussis*, more commonly known as whooping cough). In such cases, an infected person develops immunity, which he or she ultimately loses, thereby becoming susceptible to the disease again. Figure 2(a) plots infected population versus time, showing clear broad differences between the curves when the disease strength is under and above the epidemic threshold (according to Equation 1). In particular, as Figure 2(b) shows, the final epidemic size changes abruptly exactly at our predicted tipping-point (i.e. when $\lambda_1 C = 1$).

FAST IMMUNIZATION

Consider the problem of prevention of hospital-to-hospital transfer of drug resistant bacteria. Critically ill patients are frequently and routinely transferred between hospitals in order to provide necessary specialized care. While such inter-hospital transfers are an essential part of routine patient care, they also enable the transfer from hospital to hospital of highly virulent microorganisms resistant to many or all antibiotics. So, giv-

Through the emergence of big data, we can develop and test increasingly abstracted models on larger and larger sample sizes.

en a fixed amount of medicines with partial impact, like bottles of disinfectant, how should they be distributed among hospitals?

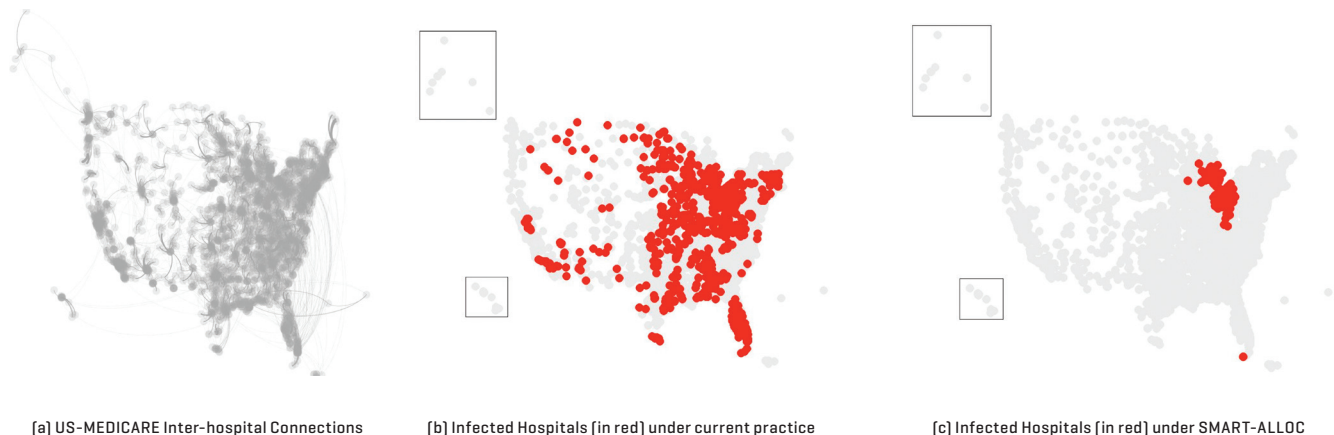
Due to the scale of this problem, any method linear in the size of the graph is better. However, an easily parallelizable algorithm would be even better. Since doctors may have different resources, each with different effectiveness, clinicians want to get good allocations quickly so that a coherent policy can be constructed and deployed. At the same time, the policy should not sacrifice accuracy. The current practice in allocating varying amounts of antidote across a network is essentially uniform, with hospitals independently tackling infection control. However, this makes no use of the connected network we are given. Another obvious method is to estimate the effect of medicines

through computer simulations. However, such simulations are computationally expensive and can often take weeks to run. Given these constraints, how can we get a practical and effective algorithm?

Collaborating with domain experts, we studied this problem and developed a fast and effective linear-time algorithm titled SMART-ALLOC [5]. Recall from our tipping-point discussion the connectivity of the network (in the form of λ_1) controls the vulnerability of a graph to an infection. Hence, we just need to decrease this value as fast as possible. It turns out that this problem is NP-hard. So, under the hood, SMART-ALLOC tries to drop the eigenvalue in a smart way. In particular, the special form of the impact function of a resource allowed us to get a provably near-optimal greedy solution. SMART-ALLOC runs in seconds on commodity hardware, as opposed to weeks required for other approaches. Figure 3 demonstrates the algorithm on the network of US-MEDICARE patient transfers.

Crucially, these results show significant benefits can be achieved by concentrating infection-control resources at a carefully chosen subset of nodes, rather than doing so in a network-agnostic fashion or using ad-hoc heuristics. The current practice has been largely focused within individual hospitals. Hence, current public-health policy is missing an opportunity to significantly reduce infection rates

Figure 3. SMART-ALLOC method has six times fewer infections (red circles). [a] The US-MEDICARE network of hospitals overlaid on a map. [b] Infected hospitals after a year (365 days) under current practice. [c] Similarly, under SMART-ALLOC. The current practice allocates equal amounts of resource to each hospital.



with an infection prevention strategy that accounts for the potential transfer of bacteria along the network of inter-hospital patient transfers.

This approach can also be extended to other scenarios, like when we can completely remove a node (i.e. vaccinate it). For example, given a large network, such as a computer communication network, which k nodes should we remove (or monitor, or immunize), to make the network as robust as possible against a computer virus attack? Making careful approximations, NETSHIELD [6] exploits the submodular structure of the set of possible solutions, getting a simple provably near-optimal algorithm.

Further, the inner-loops of both these algorithms use eigenvalue computation on graphs, which, as we have already seen earlier in this article, are very efficient to compute.

CONCLUSION

Graphs—also known as networks—are powerful tools for modeling processes and situations of interest in real-life, including social-systems, cyber-security, epidemiology, and biology. In this article we reviewed two recent developments in studying propagation-like processes on large networks: The importance of eigenvalue in understanding the tipping-point of epidemics, and subsequently leveraging that to design fast and scalable immunization policies. There are several other extensions, like having competing viruses [7] or networks that change over time [8], which we did not have space to describe here.

Really, we have given just a glimpse of the types of big-data questions we encounter after we have already built models. How can we use these models for our benefit, to actually manipulate something we care about? For example, after building models of both diseases and the underlying population, how can we study the interactions between them? How can we design policies to do effective immunization? All of these questions have to be answered in the context that we are trying to both understand and manage real-life processes on a societal-scale. These are pretty exciting times for research in networks.

References

- [1] Prakash, B. A., Chakrabarti, D., Faloutsos, M., Valler, N., and Faloutsos, C. Threshold conditions for arbitrary cascade models on arbitrary networks. In *Proceedings of the 2011 IEEE 11th International Conference on Data Mining* (Vancouver, Dec. 11-14). IEEE Computer Society, Washington, DC, 2011, 537-546.
- [2] Anderson, R. M., and May, R. M. *Infectious Diseases of Humans*. Oxford University Press, Oxford, 1991.
- [3] Kang, U. Mining Tera-Scale Graphs: Theory, engineering and discoveries. Ph.D. thesis, Carnegie Mellon University, 2012.
- [4] Eubank, S., Guclu, H., Anil Kumar, V. S., Marathe, M. V., Srinivasan, A., Toroczkai, Z., and Wang, N. Modelling disease outbreaks in realistic urban social networks. *Nature* 429, 6988 (2004), 180-184.
- [5] Prakash, B. A., Adamic, L., Iwashyna, T., Tong, H., and Faloutsos, C. Fractional immunization in networks. Under review, 2011.
- [6] H. Tong, B. A. Prakash, C. E. Tsourakakis, T. Eliassi-Rad, C. Faloutsos, and D. H. Chau. On the vulnerability of large graphs. In *Proceedings of the 2010 IEEE International Conference on Data Mining*. IEEE Computer Society, Washington, DC, 2010, 1091-1096.
- [7] Prakash, B. A., Beutel, A., Rosenfeld, R., and Faloutsos, C. Winner takes all: Competing viruses or ideas on fair-play networks. In *Proceedings of the 21st International Conference on World Wide Web* (Lyon, France, April 16-20). ACM Press, New York, 2012, 1037-1046.
- [8] Prakash, B. A., H. Tong, N. Valler, M. Faloutsos, and C. Faloutsos. Virus propagation on time-varying networks: Theory and immunization algorithms. In *Proceedings of the 2010 European Conference on Machine Learning and Knowledge Discovery in Databases: Part III* (Barcelona, Sept. 20-24). Springer-Verlag, Berlin, 2010, 99-114.

Further Reading

E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic. The role of social networks in information diffusion. In *Proceedings of the 21st International Conference on World Wide Web* (Lyon, France, April 16-20). ACM Press, New York, 2012, 1037-1046.

D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Washington, DC, Aug. 24-27). ACM Press, New York, 2003, 137-146.

J. Leskovec, L. Backstrom, and J. Kleinberg. Meme-tracking and the dynamics of the news cycle. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Paris, June 28-July 1). ACM Press, New York, 2009, 497-506.

M. Richardson and P. Domingos. Mining knowledge-sharing sites for viral marketing. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Edmonton, Canada, July 23-25). ACM Press, New York, 2002, 61-70.

Biography

B. Aditya Prakash is a Ph. D. student in the Computer Science Department at Carnegie Mellon University. He received his B. Tech. in computer science from the Indian Institute of Technology (IIT) Bombay. His research interests include data mining, applied machine learning and databases, with an emphasis on large real-world networks and time-series. He will soon be joining the Computer Science Department at Virginia Tech as an assistant professor.

© 2012 ACM 1528-4972/12/09 \$15.00



The image shows the cover of the journal 'ACM Transactions on Reconfigurable Technology and Systems'. The cover features a large, colorful diamond shape in the top right corner, composed of smaller diamonds in shades of purple, blue, and green. The title 'ACM Transactions on Reconfigurable Technology and Systems' is prominently displayed in blue and green text. Below the title, there is a list of articles from a special edition on FPGAs, including an introduction, a guest editorial, and several research papers on topics like delay variation, statistical analysis, and desktop computers with reconfigurable processors. The ACM logo and the Association for Computing Machinery name are visible at the bottom of the cover.

This quarterly publication is a peer-reviewed and archival journal that covers reconfigurable technology, systems, and applications on reconfigurable computers. Topics include all levels of reconfigurable system abstractions and all aspects of reconfigurable technology including platforms, programming environments and application successes.

www.acm.org/trets
www.acm.org/subscribe

ACM Association for Computing Machinery
 Advancing Computing as a Science & Profession