PROCEEDINGS OF THE ROYAL SOCIETY A

rspa.royalsocietypublishing.org

Research

CrossMark click for updates

Cite this article: Bui-Xuan B-M, Jones NS. 2014 How modular structure can simplify tasks on networks: parameterizing graph optimization by fast local community detection. *Proc. R. Soc. A* **470**: 20140224. http://dx.doi.org/10.1098/rspa.2014.0224

Received: 19 March 2014
Accepted: 15 July 2014

Subject Areas:
complexity, theory of computing

Keywords:
network analysis, parametrized complexity, community structure

Author for correspondence:
Nick S. Jones
e-mail: nick.jones@imperial.ac.uk

Royal Society Publishing

How modular structure can simplify tasks on networks: parameterizing graph optimization by fast local community detection

Binh-Minh Bui-Xuan[1] and Nick S. Jones[2]

[1]Laboratoire d'Informatique de Paris 6, Université Pierre et Marie Curie, Paris, France
[2]Department of Mathematics, Imperial College London, Queen's Gate, London, UK

By considering the task of finding the shortest walk through a Network, we find an algorithm for which the run time is not as $O(2^n)$, with n being the number of nodes, but instead scales with the number of nodes in a coarsened network. This coarsened network has a number of nodes related to the number of dense regions in the original graph. Since we exploit a form of local community detection as a preprocessing, this work gives support to the project of developing heuristic algorithms for detecting dense regions in networks: preprocessing of this kind can accelerate optimization tasks on networks. Our work also suggests a class of empirical conjectures for how structural features of efficient networked systems might scale with system size.

1. Introduction

(a) Networks and communities

The past decade has seen a widespread appreciation that networks in Nature have a structure which makes them poorly modelled as samples from the more traditional random graph ensembles [1]. A feature of many real networks is that a marked modular (or community) structure is present [2–6], where a network community is a subset of nodes with relatively dense connections within the subset but sparse connections to the rest of the network. Though algorithms for detecting communities are actively constructed in physics, mathematics, engineering and computer science, the reason for detecting these dense regions is not always articulated. Studies of empirical graphs suggest that nodes in the same community tend to have similar properties in the world and thus community detection can help us, for example, assign functional labels to uncharacterized nodes [2,3]. In this paper, we investigate how community detection can also help simplify problems on graphs.

Even though community detection tasks can be hard [7,8] experience with greedy algorithms suggests that plausible solutions can be found quickly for networks with a pronounced community structure (though, of course, sub-optimal solutions need to be treated with care [9]) [2,3]. Recent theoretical work in network physics and computer science also suggests that for certain types of graphs, community detection need not be costly [10–13]. It is also the case that a marked community structure is present in many empirical networks and that networks with similar functions appear to have similar community structure [4–6]. It is possible that this similarity occurs because community structure constrains dynamics on the graphs. Indeed it has been found that particular choices of dynamics on networks can in turn correspond to particular methods for detecting communities in graphs [14]. The literature asking why the networks we observe in the world are modular is substantial [15,16]: one might speculate that assembly rules for real networks are such as to simplify either optimization tasks or dynamics on them and so in turn this leads to a pronounced community structure. The notion that networks in nature are often optimized for transport is an established part of theoretical biological physics (e.g. [17,18]).

(b) Parametrized complexity

Weakly coupled to the stream of empirically motivated networks literature is recent work in computational complexity called parametrized complexity, or fixed parameter tractability [19,20]. The concerns of this vibrant field are common to the empirical interests of the network research community: how do (parametrized) constraints on graph structure simplify problems on graphs? Researchers in empirical studies of networks ask: how are real-world networks simple? Asking whether some hard problems are simple on empirical graphs is thus natural. In the following, though we use recent work from parametrized complexity, we will not be providing algorithms that have computational cost scaling in polynomial time with network size; instead we will show that a problem scaling like 2^n (n is the number of nodes in the graph) can be converted into one which scales like $\leq 2^{\tilde{n}}$ but where we can suggest $\tilde{n} \ll n$ by relating $\tilde{n}$ to (a local version of) the number of communities in the graph.

(c) Hamiltonian walk and communities

Motivated by the above observations, we ask whether a particular problem, Hamiltonian Walk, HAMWALK (an NP-hard task[1]) is simpler on networks with pronounced community structure. We define a Hamiltonian Walk as a shortest closed walk on a graph which visits every node at least once[2] [22,23]. The study of self-avoiding walks on lattices and fractals is an established area of probability and statistical physics and modifications which allow limited self-crossings have been considered [24,25]; problems of this kind are of broad relevance to understanding percolation and polymer phenomena. We note, of course, that the interface between problems in computational complexity and statistical physics is now a lively one [26,27].

We hypothesize that partitioning graphs into communities, coarsening the graphs to yield smaller graphs with nodes representing entire communities [28], and then solving problems on the coarsened graph and on the individual communities of the full graph in combination, might lead to significant computational speed-ups for some real graphs and appropriately chosen optimization tasks (figure 1). We hope to exploit the fact, noted above, that empirical networks often have pronounced community structure and finding this structure need not be hard.

[1] It is straightforward to recast HAMWALK as a travelling salesman problem which can be solved in time $O(2^n)$ [21].

[2] A Hamiltonian walk sometimes means alternatively a covering walk which is strictly self-avoiding.

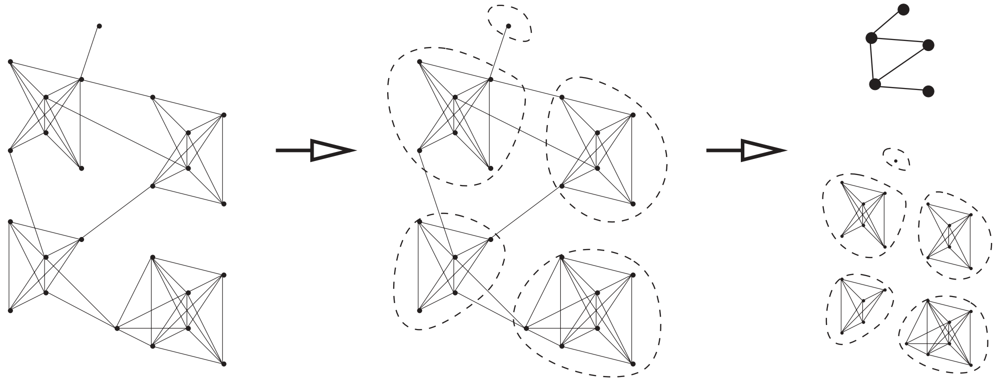

Figure 1. Community detection can be fast if pronounced communities are present [10–12]. Solutions to optimization problems on the coarsened graph can sometimes be converted into solutions for the full graph.

For real networks, this hypothesis could be converted into a class of heuristic algorithms for solving optimization problems. It thus seems desirable to see whether the preceding intuition can be expressed mathematically: is there a class of (crudely) realistic graphs for which HAMWALK has a runtime scaling which is provably less than the $O(2^n)$ bound that can currently be achieved (see footnote 1)? We believe this both motivates further work in providing tighter runtime bounds for more realistic graphs, justifies the development of appropriate heuristic optimization algorithms and makes a connection between parametrized complexity and network empirics: making explicit the notion that the modular network structure we observe in Nature could help simplify tasks for networked systems.

2. Solving Hamiltonian walk by graph coarsening

(a) A local clustering algorithm

Lokshtanov & Marx [10] study the runtime of finding partitions of networks into clusters (disjoint sets of nodes) where each cluster, C_i (for all i), has (i) a total number of links connected to nodes not in C_i that is $\leq \delta$ (call this the degree of C_i) and (ii) the number of pairs of nodes in C_i between which there is no link is $\leq \mu$. This bears some resemblance to the local community detection in [29] and seeks to identify sets of densely connected nodes (μ small) which are isolated from other such sets (δ small). If δ is treated as a fixed input then, remarkably, this problem can be solved in randomized time $2^{O(\mu)}n^{O(1)}$. Similarly, if μ is held fixed then the run time is $2^{O(\delta)}n^{O(1)}$ [10].

(b) Special cases

We first run an algorithm that detects all clusters in graph, G, with degree $\leq \delta$ and $\leq \mu$ missing links [10]. In the simple case when $\delta = 2$ and $\mu = 0$, a naive solution is as follows. Define a coarsened version of G, G', in which, for each cluster taken consecutively (the order is irrelevant), all nodes of the cluster are removed and substituted for a single node, called a cluster-node, which is connected to the nodes ($\leq$2 nodes) to which the cluster was previously connected. We then solve HAMWALK on G' using the Held-Karp algorithm (see footnote 1) [21] and obtain a walk (in G') as a result. We finally expand this walk to become a walk in G simply by a replacement of every cluster-node by its original cluster of G and quickly computing appropriate paths through the clique (note that $\mu = 0$ so this task is simple). It is a standard exercise to check the obtained walk is indeed a Hamiltonian walk of G. The intuition behind is that sets of nodes which are strongly isolated from the rest of the graph (like in the case of $\delta = 2$) allow marked simplifications of problems on the graph. This provokes the question that we consider in the following: does this intuition hold for richer classes of graphs, with more general δ and μ?

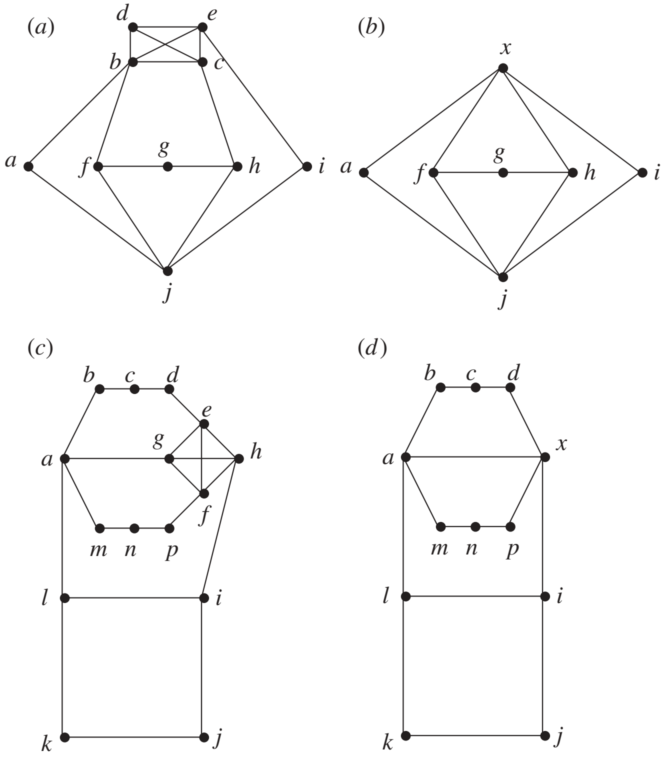

Figure 2. (*a*) Graph G, where $\{b, c, d, e\}$ is a clique. (*b*) Coarsened graph G' where clique $\{b, c, d, e\}$ is replaced by cluster-node x. (*c*,*d*) a further example of original graph G and coarsened graph G'.

Unfortunately, unlike the previous case, not all solutions to HAMWALK(G') can be easily modified to make solutions to HAMWALK(G). Consider the graph G and its coarsened graph G' in figure 2*a*,*b*. Both $W_1' = (a, x, f, g, h, x, i, j, a)$ and $W_2' = (a, j, f, g, h, j, i, x, a)$ are Hamiltonian walks of G'. Here, W_1' can be expandable to a Hamiltonian walk of G just by replacing the first occurrence of x with b and the second occurrence of x with c, d, e, namely to obtain $W_1 = (a, b, f, g, h, c, d, e, i, j, a)$ (a cycle, hence optimal in size). However, applying such local expansions on W_2' will not be as successful (not a cycle, because of multiple occurrence of j, hence, longer walk than before).

Given the above one might conclude that a careful counting of the number of times that each cluster-node is visited will help us extend, simply, our walks on G' to walks on G. In the preceding example W_1' visits x twice, whereas W_2' visits x only once, so it could be that the number of visits to the cluster-node x is important. However, graphs exist which allow two solutions to HAMWALK(G') which visit cluster-nodes the same number of times but which do not both allow a simple expansion to a solution for G (e.g. figure 2*c*,*d*). In G' (figure 2*d*), both $W_1' = (a, b, c, d, x, i, j, k, l, a, m, n, p, x, a)$ and $W_2' = (a, b, c, d, x, i, j, k, l, i, x, p, n, m, a)$ are Hamiltonian walks of G' visiting cluster-node x twice. W_1' can be locally expanded to a Hamiltonian walk of G by replacing the first occurrence of x with e, h and the second occurrence of x with f, g: $W_1 = (a, b, c, d, e, h, i, j, k, l, a, m, n, p, f, g, a)$ is a Hamiltonian walk of G. One can check that W_2' cannot be expanded to a Hamiltonian walk of G only by replacing occurrences of x with vertices from $\{e, f, g, h\}$ (the edge (h, i) must be traversed twice).

(c) Solving HAMWALK with parameters δ and μ

We now consider the case when δ and μ are unspecified positive integers, we proceed as follows. Each ith cluster, C_i, will contain a set of shell nodes, S_i, which are defined as being in C_i and either have a link to a node which is not in C_i, or lack a link to a node which is in C_i. When $|C_i| > 2 \cdot |S_i|$, we define another set of nodes, called second shell nodes, T_i, which are taken randomly from $C_i \setminus S_i$ so that $|S_i| = |T_i|$ ($|S_i| \leq \delta + 2\mu$). We call nodes which are in each detected cluster but which are neither shell nodes nor second shell nodes, good bulk nodes, $\mathrm{GB}_i = C_i \setminus S_i \setminus T_i$. GB_i is thus a clique; it is this simple structure we will exploit in the following. The set T_i is a device which will help simplify our proof.

We now define a coarsened version of G, G', in which, for each cluster with degree $\leq \delta$, and at most μ missing links and $|C_i| > 2 \cdot |S_i|$, all nodes in the good bulk are removed and substituted for a single node, called the coarsened good bulk node, b_i, which is connected to all of the shell and second shell nodes in the cluster. We will later discuss coarsened walks: a walk on G is coarsened to a walk on G' by identifying consecutive (or single) walker visits to nodes in GB_i and substituting them for single visits to the coarsened good bulk node b_i.

The outline of our proof is as follows: we create a coarsened graph, G', in which all nodes in each clique GB_i are represented as single node b_i and all other nodes are left untouched. We solve HAMWALK by a standard method on the coarsened graph. We show that, because of the simplicity of the cliques GB_i and, noting the properties of the shell and second shell, that this walk can be converted into a solution to HAMWALK on the original graph in a time polynomial in n. The use of the two shells will allow us to avoid problems identified in the examples above.

Claim 1: any Hamiltonian walk of G' can, with resources polynomial in network size, be converted to a Hamiltonian walk of G' that visits every coarsened good bulk node once and only once.

Proof of Claim 1. It is easy to check whether a Hamiltonian walk of G' visits every coarsened good bulk node once and only once. If this is not the case, we can repeatedly perform the following substitutions. Call b one of the coarsened good bulk nodes which is visited more than once. Consider any walk with a second visit to b: it is either of the form (A) $s_i b s_i$ or $s_i b s_j$ $i \neq j$, where s_i and s_j are nodes in the shell of the cluster to which b belongs, or (B) $t_i b t_i$ or $t_i b t_j$ $i \neq j$, where t_i and t_j are second shell nodes, or (C) $s_i b t_j$ (or $t_i b s_j$), where s_i is a shell node and t_j a second shell node. Now we can obtain another closed walk that still visits every node of G' at least once, b included, as follows: (A) replace $s_i b s_i$ by s_i, or replace $s_i b s_j$ by $s_i t s_j$, where t is any node from the second shell; (B) replace $t_i b t_i$ by t_i, or replace $t_i b t_j$ by $t_i t_j$ and (C) replace $s_i b t_j$ by $s_i t_j$. In all cases, the length of the modified walk is not increased, that is, if the original is a Hamiltonian walk then the modified walk is still a Hamiltonian walk of G'. ■

Given any Hamiltonian walk of G', we can thus obtain a Hamiltonian walk of G' having the property described in *Claim 1*. Denote its length by $w_{G'}$. We can un-coarsen the walk into a walk on G by locally extending using a greedy approach at every good bulk node. What *Claim 1* implies is that the extended walk has length $w_{G'} + \sum_i (|\mathrm{GB}_i| - 1)$ (by exploiting the clique structure of GB_i). Clearly, $w_{G'} + \sum_i (|\mathrm{GB}_i| - 1) \geq w_G$, where w_G is the length of Hamiltonian walks of G. Now, if this were an equality, we would have shown the following.

Main claim: any solution to HAMWALK(G') can, with resources polynomial in network size, be converted into solutions of HAMWALK(G).

Claim 2: there is always a Hamiltonian walk of G which, when coarsened to a walk on G', visits coarsened good bulk nodes once and only once.

The *Main claim* is true provided that *Claim 2* is true. This is because it would imply that $w_{G'} + \sum_i (|\mathrm{GB}_i| - 1)$ is equal to w_G (pick the Hamiltonian walk given by *Claim 2*; consider its coarsening on G' whose length has to be $w_G - \sum_i (|\mathrm{GB}_i| - 1)$; but this is also a closed walk of G' that visits every node of G' at least once and thus is of length $\geq w_{G'}$. We know from *Claim 1* that $w_{G'} + \sum_i (|\mathrm{GB}_i| - 1) \geq w_G$ and hence the *Main claim* follows).

Proof of Claim 2. Clusters are pairwise independent, so that we will only give a proof for *Claim 2* with respect to a particular cluster C, with set of shell nodes S, set of second shell nodes T and set of good bulk nodes GB. Among the Hamiltonian walks of G, consider one that makes distinct visits to nodes in GB the least number of times (a distinct visit to GB is a part of a walk containing a contiguous sequence of nodes in GB immediately preceded and followed by a visit to nodes not in GB). By contradiction suppose this walk contains repeated distinct visits to GB. Consider visits to $T \cup \mathrm{GB}$: because nodes outside C can connect to $T \cup \mathrm{GB}$ only via the shell S, such a visit has to be of the form $s_i w s_j$, where s_i and s_j are shell nodes and w a walk in $T \cup \mathrm{GB}$. Denote by p the number of times $T \cup \mathrm{GB}$ is visited: some thought shows that p is no greater than $|S|$ and hence not greater than $|T|$. Now replace the first $p-1$ visits to $T \cup \mathrm{GB}$ keeping the same shell nodes but the walk in $T \cup \mathrm{GB}$ is substituted for a random (chosen without reuse) second shell node: for instance, $s_i w s_j$ is substituted for $s_i t_k s_j$ with some $t_k \in T$. Denote the last visit to $T \cup \mathrm{GB}$ by $s_k u s_l$ (this includes the $p=1$ case) and substitute it for $s_k v s_l$, where v contains all nodes in $T \cup \mathrm{GB}$ (except for those of T that have been used for the $p-1$ previous visits) and v visits all nodes of GB contiguously. We now consider the result of these substitutions: clearly, this is still a closed walk that visits every node of G at least once, furthermore, it has the same length (or less) as the original walk before substitutions. In other words, we have obtained a Hamiltonian walk of G, but one that visits nodes in GB only once: a contradiction. ■

(d) Algorithm solving HAMWALK

Given μ and δ (where either μ or δ, but not both, might be a function of n) take G on n nodes and create the graph G' using [10]. In the case that μ is fixed this takes a time $n^{O(1)}2^{O(\delta)}$ [10]. Calculate the matrix of shortest paths for G', and use the Held-Karp algorithm to find a solution to HAMWALK(G') [21]. Convert it into one having the property described in *Claim 1*. Locally extend this to a walk in G using a greedy substitution at every coarsened good bulk node. Return the latter as a solution to HAMWALK(G). Runtime: $2^{(2\delta+4\mu+1)n_\mathrm{c}+n'} + n^{O(1)}2^{O(\delta)}$ where n_c is the number of clusters with degree $\leq \delta$, at most μ missing links, and $|C| > 2 \cdot |S|$ and n' is the number of nodes which are not in clusters with $|C| > 2 \cdot |S|$ (note $|S| \leq \delta + 2\mu$).

(e) A note on clique-width

Clique-width is a quantity which has proved very popular in the area of parametrized complexity: it is related to a natural approach to assembling a graph [30]. We first define clique-width and then note some implications of our work (thereby illustrating its relevance in parametrized complexity). We suppose a k-graph has nodes with labels from the set $\{1, 2, \ldots, k\}$. We define a seed k-graph with one node and a label from this set. The clique-width of a graph G is the smallest integer k such that G can be composed by repeated use of four simple operations: (i) generate: make a seed k-graph labelled by i, (ii) disjoint union: two distinct graphs are treated as disconnected components of the same graph, (iii) combine: linking all nodes with label j with nodes labelled i, (iv) relabel: all indices i replaced with j. This protocol can generate graphs which contain cliques. In this paper, clique-width can be connected to the case where there are no missing links inside the cliques identified ($\mu = 0$). In this case, it can be proved fairly easily that the clique-width of the graph is $\leq k = (\delta n_\mathrm{c} + n')$ and a corresponding protocol for constructing the graph using the above four operations can be provided (this witness protocol, demonstrating that the bound can be met, is called a k-expression). Given this protocol, a number of Monadic Second Order Logic (MSOL) problems can be solved (see [31] and the numerous MSOL problems therein) including the classic problem 'Minimum Dominating Set' [32] in a time $2^{O(\delta n_\mathrm{c}+n')}$. This last follows from the recent observation that, given a k-expression for clique-width k, Minimum Dominating Set can be solved in a time 2^{2k} [33]. Thus, our result for HAMWALK can be extended to other problems.

3. Discussion and conclusion

We have proved that it is possible to solve HAMWALK in time $2^{(2\delta+4\mu+1)n_\mathrm{c}+n'} + n^{O(1)}2^{O(\delta)}$. Some care is required in the interpretation of this runtime. While δ and μ are parameters of the algorithm (with forms which can be specified independent of the graph) by contrast n_c is a feature of the graph for a given δ and μ. Despite this, one can hope to construct graphs which have a given, n_c, n', δ and μ. To help the interpretation of this result, we thus consider the following graph family: every node is inside one and only one of n_c clusters such that for each cluster, μ is constant, $2^{O(\delta)} = n^{O(1)}$, $n_c\delta \leq O(1)\log n$ (for this family $n' = 0$) and each cluster is of size $|C| > 2(\delta + 2\mu)$. This is a graph family composed of dense clusters in which links between the clusters and missing links inside the clusters are proportionately rare (an equivalent family with the roles of μ and δ reversed could also be considered): as graphs increase in size the number of clusters is relatively slow growing. HAMWALK can be solved on this family in time $n^{O(1)}$ (if the values of μ and δ are not known in advance, finding appropriate choices only yields a polynomial time overhead). In the intuitive case with μ and δ both constant, the number of clusters increases as the logarithm of the number of nodes. Our abstract complexity based argument can be used to inspire a class of empirical conjectures about optimal growing networks in Nature. Suppose that, in order to thrive, a network in the world has to solve HAMWALK on itself efficiently for larger and larger system sizes. Given the above, we might thus hypothesize that, if we have observations of the system at a variety of network sizes, we will find that the number of communities would increase like the logarithm of the number of nodes. While few natural systems are likely optimized to solve HAMWALK, this form of reasoning might allow us to relate the system-size scaling of structural features to the tasks networked-systems are optimized to solve. Why natural networks might show modular structure is a canonical question; beyond conjectured roles for communities like helping networked systems to control their dynamics, or to be more evolvable [15,16], we have provided a concrete setting in which community structure helps simplify optimization tasks: while the presence of modular structure might typically be seen as a signature of purely local, module-level, computations it could also be a signature of an efficient approach to global optimization.

The family specified above was selected to share some similarities with real networks, while remaining mathematically tractable: many real networks do have sets of densely connected nodes which are relatively isolated from the rest of the network. In the preceding proof, we apparently exploited the clique structure of the good bulk and second shell; we note that this is not a necessary condition for our basic proof structure to work and, at least, small numbers of missing links can be tolerated within the good bulk. Widening our results to less constrained classes of graphs, and so to alternative definitions of communities, is a natural extension of our work. The bound, δ, gives some indication of the modular nature of the graph: for graphs with appropriately small δ the clusters are more isolated from each other and it is easier to solve HAMWALK; for appropriately small μ again HAMWALK is easier to solve. If $n' \ll n$, indicating a graph with many nodes inside clusters, then again it is much easier to solve HAMWALK. The bound we proved is by construction: it might be possible to solve HAMWALK in faster time. We obtained our bound on HAMWALK by coarsening the graph on n nodes and solving problems on the reduced version. It is known that empirical networks have hierarchical community structure: dense regions embedded inside others. Authors have considered renormalization approaches to such networks [28]. How this hierarchical structure simplifies problems on graphs, and how repeated coarsenings of the full graph might help is open. In this setting, not only does modular structure constrain dynamics on graphs, but it can also simplify problems on graphs. Many practical optimization problems are posed on real graphs with modular structure. The preceding gives, by a proof for a simple class of graphs, a hint that some of the optimization problems we are interested in might allow heuristics which run in time scaling with the number of communities (and parametrizations of their isolation) rather than the number of nodes: this gives a justification for the exercise of developing community detection algorithms.

Acknowledgements. With thanks to Sumeet Agarwal and also to Sam Johnson, Iain Johnston and Simone Severini.
Funding statement. N.S.J. acknowledges support from EPSRC grant nos. EP/I005986/1 and EP/I005765/1.

References

1. Newman M. 2010 *Networks: an introduction*. Oxford, UK: Oxford University Press.
2. Porter M, Onnela J-P, Mucha P. 2009 Communities in networks. *Not. Am. Math. Soc.* **56**, 1164–1166.

3. Fortunato S. 2010 Community detection in graphs. *Phys. Rep.* **486**, 75–174. (doi:10.1016/j.physrep.2009.11.002)
4. Lancichinetti A, Kivela M, Saramaki J, Fortunato S. 2010 Characterizing the community structure of complex networks. *PLoS ONE* **5**, e11976. (doi:10.1371/journal.pone.0011976)
5. Onnela J-P, Fenn DJ, Reid S, Porter MA, Mucha PJ, Fricker MD, Jones NS. 2012 Taxonomies of networks from community structure. *Phys. Rev. E* **86**, 036104. (doi:10.1103/PhysRevE.86.036104)
6. Agarwal S. 2012 Networks in nature: dynamics, evolution and modularity. PhD thesis, Oxford University, UK.
7. Brandes U, Delling D, Gaertler M, Görke R, Hoefer M, Nikoloski Z, Wagner D. 2008 On modularity clustering. *IEEE Trans. KDE* **20**, 172–188.
8. Newman M, Girvan M. 2004 Finding and evaluating community structure in networks. *Phys. Rev. E* **69**, 026113. (doi:10.1103/PhysRevE.69.026113)
9. Good B, de Montjoye Y-A, Clauset A. 2010 The performance of modularity maximization in practical contexts. *Phys. Rev. E* **81**, 046106. (doi:10.1103/PhysRevE.81.046106)
10. Lokshtanov D, Marx D. 2013 Clustering with local restrictions. *Inf. Comput.* **222**, 278–292. (doi:10.1016/j.ic.2012.10.016)
11. Decelle A, Krzakala F, Moore C, Zdeborova L. 2011 Inference and phase transitions in the detection of modules in sparse networks. *Phys. Rev. Lett.* **107**, 065701. (doi:10.1103/PhysRevLett.107.065701)
12. Decelle A, Krzakala F, Moore C, Zdeborova L. 2011 Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Phys. Rev. E* **84**, 066106. (doi:10.1103/PhysRevE.84.066106)
13. Nadakuditi R, Newman M. 2012 graph spectra and the detectability of community structure in networks. *Phys. Rev. Lett.* **108**, 188701. (doi:10.1103/PhysRevLett.108.188701)
14. Lambiotte R, Delvenne J-C, Barahona M. 2008 Laplacian dynamics and multiscale modular structure in networks. (http://arxiv.org/abs/0812.1770).
15. Simon H. 1962 The architecture of complexity. *Proc. Am. Phil. Soc.* **106**, 467–482.
16. Meunier D, Lambiotte R, Bullmore E. 2010 Modular and hierarchically modular organization of brain networks. *Front. Neurosci.* **4**, 200. (doi:10.3389/fnins.2010.00200)
17. West G, Brown J, Enquist B. 1997 A general model for the origin of allometric scaling laws in biology. *Science* **276**, 122–126. (doi:10.1126/science.276.5309.122)
18. Katifori E, Szollosi G, Magnasco M. 2010 Damage and fluctuations induce loops in optimal transport networks. *Phys. Rev. Lett.* **104**, 048704. (doi:10.1103/PhysRevLett.104.048704)
19. Downey R, Fellows M. 1999 *Parameterized complexity*. New York: Springer.
20. Niedermeier R. 2006 *Invitation to fixed-parameter algorithms*. Oxford, UK: Oxford University Press.
21. Held M, Karp R. 1962 A dynamic programming approach to sequencing problems. *J. Soc. Indust. Appl. Math.* **10**, 196–210. (doi:10.1137/0110015)
22. Goodman S, Hedetniemi S. 1974 On Hamiltonian walks in graphs. *SIAM J. Comp.* **3**, 214–221. (doi:10.1137/0203017)
23. Bermond J-C. 1975 On Hamiltonian walks. In *Proc. 5th British Comb. Conference* (eds CStJA Nash-Williams, J Sheehan), pp. 41–51. Winnipeg, Canada: Utilitas Math., Publ. Inc.
24. Madras N, Slade G. 1996 *The self-avoiding walk*. Boston, MA: Birkhäuser.
25. van der Hofstad R, König W. 2001 A survey of one-dimensional random polymers. *J. Stat. Phys.* **103**, 915–944. (doi:10.1023/A:1010309005541)
26. Mézard M, Montanari A. 2009 *Information, physics, and computation*. Oxford, UK: Oxford University Press.
27. Moore C, Mertens S. 2011 *Nature of computation*. Oxford, UK: Oxford University Press.
28. Radicchi F, Ramasco J, Barrat A, Fortunato S. 2008 Complex networks renormalization: flows and fixed points. *Phys. Rev. Lett.* **101**, 148701. (doi:10.1103/PhysRevLett.101.148701)
29. Clauset A. 2005 Finding local community structure in networks. *Phys. Rev. E* **72**, 026132. (doi:10.1103/PhysRevE.72.026132)
30. Courcelle B, Olariu S. 1997 Upper bounds to the clique-width of graphs. *Discrete Appl. Math.* **101**, 77–114. (doi:10.1016/S0166-218X(99)00184-5)
31. Courcelle B, Engelfriet J. 2012 *Graph structure and monadic second-order logic, a language theoretic approach*. Cambridge, UK: Cambridge University Press.
32. Garey M, Johnson D. 1979 *Computers and intractability: a guide to the theory of NP-completeness*. New York, NY: W. H. Freeman.
33. Bodlaender HL, Leeuwen EJV, Rooij JMMV, Vatshelle M. 2010 Faster algorithms on branch and clique decompositions. *Math. Found. Comput. Sci.* **6281**, 174–185.