Connected Coloring Completion for General Graphs: Algorithms and Complexity*

Benny Chor[1], Michael Fellows[2,3], Mark A. Ragan[4], Igor Razgon[5], Frances Rosamond[2], and Sagi Snir[6]

[1] Computer Science Department, Tel Aviv University, Tel Aviv, Israel
benny@cs.tau.ac.il
[2] University of Newcastle, Callaghan NSW 2308, Australia
{michael.fellows,frances.rosamond}@newcastle.edu.au
[3] Durham University, Institute of Advanced Study, Durham DH1 3RL, United Kingdom
[4] Institute for Molecular Biosciences, University of Queensland, Brisbane, QLD 4072 Australia
m.ragan@imb.uq.edu.au
[5] Computer Science Department, University College Cork, Ireland
i.razgon@cs.ucc.ie
[6] Department of Mathematics, University of California, Berkeley, USA
ssagi@math.berkeley.edu

Abstract. An *r-component connected coloring* of a graph is a coloring of the vertices so that each color class induces a subgraph having at most r connected components. The concept has been well-studied for $r = 1$, in the case of trees, under the rubric of *convex coloring*, used in modeling perfect phylogenies. Several applications in bioinformatics of connected coloring problems on general graphs are discussed, including analysis of protein-protein interaction networks and protein structure graphs, and of phylogenetic relationships modeled by splits trees. We investigate the r-COMPONENT CONNECTED COLORING COMPLETION (r-CCC) problem, that takes as input a partially colored graph, having k uncolored vertices, and asks whether the partial coloring can be completed to an r-component connected coloring. For $r = 1$ this problem is shown to be NP-hard, but fixed-parameter tractable when parameterized by the number of uncolored vertices, solvable in time $O^*(8^k)$. We also show that the 1-CCC problem, parameterized (only) by the treewidth t of the graph, is fixed-parameter tractable; we show this by a method that is of independent interest. The r-CCC problem is shown to be $W[1]$-hard, when parameterized by the treewidth bound t, for any $r \geq 2$. Our proof also shows that the problem is NP-complete for $r = 2$, for general graphs.

Topics: Algorithms and Complexity, Bioinformatics.

1 Introduction

The following two problems concerning colored graphs can be used to model several different issues in bioinformatics.

* This research has been supported by the Australian Research Council through the Australian Centre in Bioinformatics. The second and fifth authors also acknowledge the support provided by a William Best Fellowship at Grey College, Durham, while the paper was in preparation.

G. Lin (Ed.): COCOON 2007, LNCS 4598, pp. 75–85, 2007.

r-Component Connected Recoloring (r-CCR)
Instance: A graph $G = (V, E)$, a set of colors $\mathcal{C}$, a coloring function $\Gamma : V \rightarrow \mathcal{C}$, and a positive integer k.
Parameter: k
Question: Is it possible to modify Γ by changing the color of at most k vertices, so that the modified coloring Γ' has the property that each color class induces a subgraph with at most r components?

In the case where G is a tree and $r = 1$, the problem is of interest in the context of maximum parsimony approaches to phylogenetics [17,13]. A connected coloring corresponds to a perfect phylogeny, and the recoloring number can be viewed as a measure of distance from perfection. The problem was introduced by Moran and Snir, who showed that Convex Recoloring for Trees (which we term 1-CCR) is NP-hard, even for the restriction to colored paths. They also showed that the problem is fixed-parameter tractable, and described an FPT algorithm that runs in time $O(k(k/\log k)^k n^4)$ for colored trees [17]. Subsequently, Bodlaender *et al.* have improved this to an FPT algorithm that runs in linear time for every fixed k, and have described a polynomial-time kernelization to a colored tree on at most $O(k^2)$ vertices [3].

Here we study a closely related problem.

r-Component Connected Coloring Completion (r-CCC)
Instance: A graph $G = (V, E)$, a set of colors $\mathcal{C}$, a coloring partial function $\Gamma : V \rightarrow \mathcal{C}$ where there are k uncolored vertices.
Parameter: k
Question: Is it possible to complete Γ to a total coloring function Γ' such that each color class induces a subgraph with at most r components?

The problem is of interest in the following contexts.

(1) Protein-protein interaction networks. In a protein-protein interaction network the vertices represent proteins and edges model interactions between that pair of proteins [22,7,20,21]. Biologists are interested in analyzing such relationship graphs in terms of cellular location or function (either of which may be represented by vertex coloring) [16]. Interaction networks colored by cellular location would be expected to have monochrome subgraphs representing localized functional subnetworks. Conversely, interaction networks colored by function may also be expected to have monochrome connected subgraphs representing cellular localization. The issue of error makes the number of recolorings (corrections) needed to attain color-connectivity of interest [17], and the issue of incomplete information may be modeled by considering uncolored vertices that are colored to attain color-connectivity, the main combinatorial problem we are concerned with here. Protein-protein interaction graphs generally have bounded treewidth.

(2) Phylogenetic networks. Phylogenetic relationships can be represented not only as trees, but also as networks, as in the *splits trees* models of phylogenetic relationships that take into account such issues as evidence of lateral genetic transfer, inconsistencies in the phylogenetic signal, or information relevant to a specific biological hypothesis, e.g., host-parasite relationships [15,14]. Convex colorings of splits trees have essentially the same modeling uses and justifications as in the case of trees [17,4,13]. Splits trees for natural datasets have small treewidth.

Our main results are summarized as follows:

1. 1-CCC is NP-hard for general colored graphs, even if there are only two colors.
2. 1-CCC for general colored graphs, parameterized by the number k of uncolored vertices, is fixed-parameter tractable, and can be solved in time $O^*(8^k)$.
3. 1-CCC is in XP for colored graphs of treewidth at most t, parameterized by t. (That is, it is solvable in polynomial time for any fixed t. Note that under this parameterization the number of uncolored vertices is unbounded.)
4. 1-CCC is fixed-parameter tractable when parameterized by treewidth.
5. For all $r \geq 2$, r-CCC, parameterized by a treewidth bound t, is hard for $W[1]$.

For basic background on parameterized complexity see [10,11,19].

2 Connected Coloring Completion Is NP-Hard

Theorem 1. *The 1-CCC problem is NP-hard, even if there are only two colors.*

Proof. (Sketch.) The reduction from 3SAT can be inferred from Figure 1 (the details are omitted due to space limitations). The two colors are T and F.

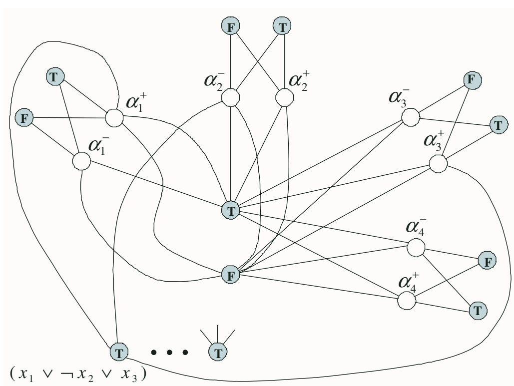

Fig. 1. The reduction from 3SAT

3 1-CCC for k Uncolored Vertices Is Fixed-Parameter Tractable

The input to the problem is a partially colored graph $G = (V, E)$, and the parameter is the number of uncolored vertices.

Soundness for the following reduction rules is easy to verify.

Rule 1. A maximal connected monochromatic subgraph (of colored vertices) can be collapsed to a single vertex. The parameter is unchanged.
Rule 2. If a color occurs on only a single vertex, then that vertex can be deleted. The parameter is unchanged.
Rule 3. An edge between two colored vertices of different color can be deleted. The parameter is unchanged.

Suppose that a partially colored graph G is reduced with respect to the above three reduction rules. The situation can be represented by a bipartite *model graph* that on one side (let us say, the *left side*), has vertices representing the vertices created by Rule 1, but not deleted by Rule 2. On the *right side* are the k uncolored vertices (and their adjacencies), and between the two sides are edges that represent an incidence relationship. Clearly, if in this representation, there are more than k vertices on the left, then the answer is NO. Thus, there are at most k colors represented on the left, and an FPT algorithm that runs in time $O^*(k^k)$ follows by exploring all possibilities of coloring the k uncolored vertices (on the right) with the k colors represented on the left of the model. We can do better than this.

Theorem 2. *1-CCC parameterized by the number of uncolored vertices is fixed-parameter tractable, solvable in time $O^*(8^k)$.*

Proof. We use the *model graph* described above. Instead of the brute-force exploration of all possibilities of coloring the right-side vertices with the left-side colors, we employ a dynamic programming algorithm. Let H denote the set of at most k uncolored vertices, and let $\mathcal{C}$ denote the set of colors represented on the left of the model graph. Create a table of size $2^{\mathcal{C}} \times 2^{H}$. We employ a table T to be filled in with 0/1 entries. The entry $T(\mathcal{C}', H')$ of the table T indexed by $(\mathcal{C}', H')$, where $\mathcal{C}' \subseteq \mathcal{C}$ and $H' \subseteq H$, represents whether (1) or not (0), it is possible to solve the connectivity problem for the colors in $\mathcal{C}'$ by assigning colors to the vertices of H'. (The only way this can happen, is that, for each color $c \in \mathcal{C}'$, the (single) connected component of c-colored vertices on the right, dominates all the c-colored vertices on the left.) We have the following recurrence relationship for filling in this table:

$T(\mathcal{C}', H') = 1 \iff \exists(c, H''),\ c \in C',\ H'' \subset H'$, such that $T(\mathcal{C}' - c, H'') = 1$ and $H' - H''$ induces a connected subgraph that dominates the vertices of color c. The table T has at most $2^k \times 2^k = 4^k$ entries, and computing each entry according to the recurrence requires time at most $O(k \cdot 2^k)$, so the total running time of the dynamic programming algorithm is $O^*(8^k)$.

4 Bounded Treewidth

Most natural datasets for phylogenetics problems have small bounded treewidth. 1-CCR is NP-hard for paths (and therefore, for graphs of treewidth one) [17]. Bodlaender and Weyer have shown that 1-CCR parameterized by (k, t), where k is the number of vertices to be recolored, and t is a treewidth bound, is fixed-parameter tractable, solvable in linear time for all fixed k [6].

4.1 1-CCC Parameterized by Treewidth is Linear-Time FPT

We describe an algorithm for the 1-CCC problem that runs in linear time for any fixed treewidth bound t, and we do this by using the powerful machinery of Monadic Second Order (MSO) logic, due to Courcelle [9] (also [1,5]). At first glance, this seems either surprising or impossible, since MSO does not provide us with any means of describing families of colored graphs, where the number of colors is unbounded. We employ a "trick" that was first described (to our knowledge) in a paper in these proceedings [3]. Further applications of what appears to be more a useful new method, rather than just a trick, are described in [12].

The essence of the trick is to construct an auxiliary graph that consists of the original input, augmented with additional *semantic vertices*, so that the whole ensemble has — or can safely be assumed to have — bounded treewidth, and relative to which the problem of interest *can* be expressed in MSO logic.

Let $G = (V, E)$ be a graph of bounded treewidth, and $\Gamma : V' \rightarrow \mathcal{C}$ a vertex-coloring function defined on a subset $V' \subseteq V$. (Assume each color in $\mathcal{C}$ is used at least once.) We construct an auxiliary graph G' from G in the following way: for each color $c \in \mathcal{C}$, create a new *semantic vertex* v_c (these are all of a second *type* of vertex, the vertices of V are of the first type). Connect v_c to every vertex in G colored c by Γ.

Consider a tree decomposition Δ for G, witnessing the fact that it has treewidth at most t. This can be computed in linear time by Bodlaender's algorithm.

Say that a color $c \in \mathcal{C}$ is *relevant* for a bag B of Δ if either of the following holds:
(1) There is a vertex $u \in B$ such that $\Gamma(u) = c$. (When this holds, say that c is *present* in B.)
(2) There are bags B' and B'' of Δ such that B is on the unique path from B' to B'' relative to the tree that indexes Δ, and there are vertices $u' \in B'$ and $u'' \in B''$ such that $\Gamma(u') = \Gamma(u'') = c$, and furthermore, c is not present in B. (When this holds, say that c is *split* by the bag B.)

Lemma 1. *If the colored graph G is a yes-instance for 1-CCC, then for any bag B, there are at most $t + 1$ relevant colors.*

Proof. Suppose that a bag B has more than $t + 1$ relevant colors, and that p of these are present in B. If s denotes the number of colors split by B, then $s > t + 1 - p$. Since B contains at most $t + 1$ vertices, the number of colors split by B exceeds the number of uncolored vertices in B, and because each bag is a cutset of G, it follows that G is a no-instance for 1-CCC.

Lemma 2. *If the colored graph G is a yes-instance for 1-CCC, then the auxiliary graph G' has treewidth at most $2t + 1$.*

Proof. Consider a tree decomposition Δ for G witnessing that the treewidth of G is at most t. By the above lemma, if we add to each bag B of Δ all those vertices v_c for colors c that are relevant to B, then (it is easy to check) we obtain a tree-decomposition Δ' for G' of treewidth at most $2t + 1$.

Theorem 3. *The 1-CCC problem, parameterized by the treewidth bound t, is fixed-parameter tractable, solvable in linear time for every fixed t.*

Proof. The algorithm consists of the following steps.

Step 1. Construct the auxiliary graph G'.
Step 2. Compute in linear time, using Bodlaender's algorithm, a tree-decomposition for G' of width at most $2t + 1$, if one exists. (If not, then correctly output NO.)
Step 3. Otherwise, we can express the problem in MSO logic. That this is so, is not entirely trivial, and is argued as follows (sketch).

The vertices of G' can be considered to be of three types: (i) the original colored vertices of G (that is, the vertices of V'), (ii) the uncolored vertices of G (that is, the vertices of $V - V'$), and (iii) the color-semantic vertices added in the construction of G'. (The extension of MSO Logic to accomodate a fixed number of vertex types is routine.)

If G is a yes-instance for the problem, then this is witnessed by a set of edges F between vertices of G (both colored and uncolored) that provides the connectivity for the color classes. In fact, we can choose such an F so that it can be partitioned into classes F_c, one for each color $c \in \mathcal{C}$, such that the classes are disjoint: no vertex $v \in V$ has incident edges $e \in F_c$ and $e' \in F_{c'}$ where $c \neq c'$.

The following are the key points of the argument:

(1) Connectivity of a set of vertices, relative to a set of edges, can be expressed by an MSO formula.

(2) We assert the existence of a set of edges F of $G \subseteq G'$, and of a set of edges F' between uncolored vertices of G and color-semantic vertices of G' such that:

- Each uncolored vertex of G has degree 1 relative to F'. (The edges of F' thus represent a coloring of the uncolored vertices of G.)
- If u and v are colored vertices of G that are connected relative to F, then there is a unique color-semantic vertex v_c such that both u and v are adjacent to v_c.
- If u is a colored vertex of G and v is an uncolored vertex of G that are connected via edges in F, then there is a unique color-semantic vertex v_c such that v is adjacent to v_c by an edge of F', and u is adjacent to v_c by an edge of G'.
- If u is an uncolored vertex of G and v is an uncolored vertex of G that are connected via edges in F, then there is a unique color-semantic vertex v_c such that v is adjacent to v_c by an edge of F', and u is adjacent to v_c by an edge of F'.
- If u and v are colored vertices of G that are both adjacent to some color-semantic vertex v_c, then u and v are connected relative to F.

4.2 r-CCC Parameterized by Treewidth is $W[1]$-Hard for $r \geq 2$

In view of the fact that 1-CCC is fixed-parameter tractable for bounded treewidth, it may be considered surprising that this does not generalize to r-CCC for any $r \geq 2$.

Theorem 4. *The 2-CCC problem, parameterized by the treewidth bound t, is hard for $W[1]$.*

Proof. (Sketch.) The proof is an FPT Turing reduction, based on color-coding [2]. We reduce from the $W[1]$-hard problem of k-CLIQUE. Let $(G = (V, E), k)$ be an instance of the parameterized CLIQUE problem. Let $\mathcal{H}$ be a suitable family of hash functions $h : V \to \mathcal{A} = \{1, ..., k\}$.

If G is a yes-instance for the k-CLIQUE problem, then for at least one $h \in \mathcal{H}$, the coloring function h is injective on the vertices of a witnessing k-clique in G (that is, each vertex of the k-clique is assigned a different color).

We describe a Turing reduction to instances $G'(h)$ of the 2-CCC problem, one for each $h \in \mathcal{H}$, such that G is a yes-instance for the k-CLIQUE problem if and only at least one $G'(h)$ is a yes-instance for the 2-CCC problem. Each $G'(h)$ has treewidth $t = O(k^2)$.

The construction of $G'(h)$ is based on an *edge-representation of the clique* strategy. We will describe the construction of $G'(h)$ in stages, building up in a modular fashion. A *module* of the construction will be a subgraph that occurs in $G'(h)$ as an induced subgraph, except for a specified set of boundary vertices of the module. These boundary sets will be identified as the various modules are "plugged together" to assemble $G'(h)$. In the figures that illustrate the construction, each kind of module is represented by a symbolic *schematic*, and the modules are built up in a hierarchical fashion. Square vertices represent uncolored vertices.

Figure 2 illustrates the *Choice Module* that is a key part of our construction, and its associated schematic representation. A Choice Module has four "output" boundary vertices, labeled $c_1, ..., c_4$ in the figure. It is easy to see that the module admits a partial solution coloring that "outputs" any one of the (numbered) colors occuring in the module depicted, in the sense that the vertices $c_1, ..., c_4$ are assigned the output color (which is *unsolved*, that is, this color class is not connected in the module), and that the other colors are all solved internally to the module, in the sense that there is (locally) only one connected component of the color class.

A *Co-Incident Edge Set Module* is created from the disjoint union of $k - 1$ Choice Modules, as indicated in in Figure 3. The boundary of the module is the union of the boundaries of the constituent Choice Modules.

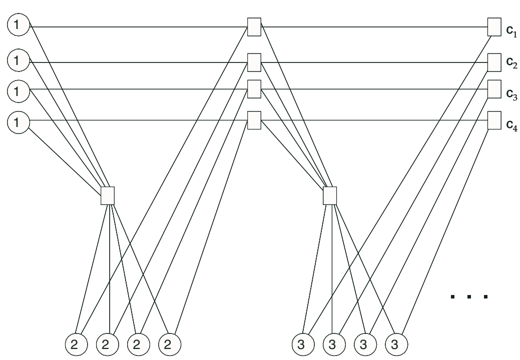

Fig. 2. A *Choice Module* of size 3

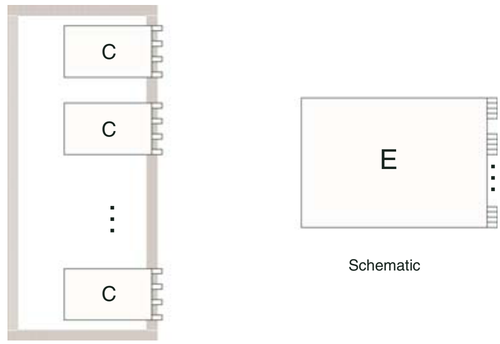

Fig. 3. A Co-Incident Edge Choice Module of size *s* is the disjoint union of *s* Choice Modules

An *XOR Stream Module* is shown in Figure 4. This has two "input" boundary sets, each consisting of four uncolored vertices, and one "output" boundary set of four uncolored vertices.

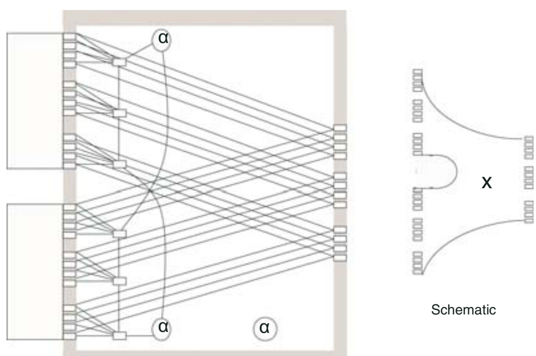

Fig. 4. An XOR Stream Module

Figure 5 shows how a *Tree of Choice Module* is assembled from Co-Incident Edge Set modules and XOR Stream modules. By the *size* of a Tree of Choice module we refer to the number of Co-Incident Edge Set modules occuring as leaves in the construction.

The overall construction of $G'(h)$ for $k = 5$ is illustrated in Figure 6. Suppose the instance graph $G = (V, E)$ that is the source of our reduction from CLIQUE has $|V| = n$ and $|E| = m$. Then each Tree of Choice module $T(h, i)$ has size $n(h, i)$, where $n(h, i)$

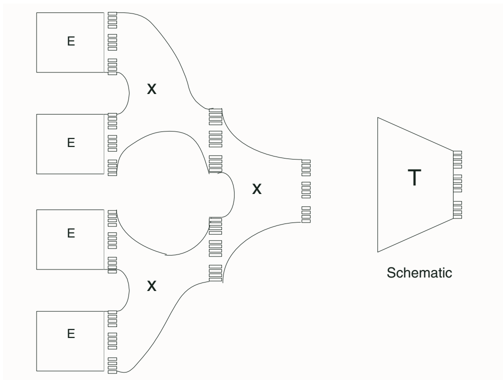

Fig. 5. A Tree of Choice Module of size 4

is the number of vertices colored $i \in \{1, ..., k\}$ by h. Let $V(h,i)$ denote the subset of vertices of V colored i by h.

In the coloring of G by h, if it should happen that a vertex v colored i has no neighbors of color j, $j \neq i$, then v cannot be part of a multicolored k-clique in G, and can be deleted. We consider only colorings of G that are *reduced* in this sense.

The "leaves" of $T(h,i)$ consist of Co-Incident Edge Set modules $E(h,i,u)$, one for each $u \in V(h,i)$. The Co-Incident Edge Set module $E(h,i,u)$ consists of $k-1$ Choice modules $C(h,i,u,j)$, $j \in \{1, ..., k\}$ and $j \neq i$, and each of these has size equal to the number of edges uv incident to u in G, where v is colored j by h.

The colors used in the construction of $E(h,i,u)$ are in 1:1 correspondence with the edges incident to v in G. Overall, the *colors* of the colored vertices in G' occuring in the Choice modules represent, in this manner, *edges* of G. Each XOR module M has three vertices colored $\alpha = \alpha(M)$ where this color occurs nowhere else in G' (see Figure 4). One of these vertices colored α is an isolated vertex.

Verification that if G has a k-clique, then G' admits a solution to the CCC problem is relatively straightforward. It is important to note that if uv is an edge of G, where $h(u) = i$ and $h(v) = j$, then the color corresponding to uv occurs in exactly two Choice modules: in $C(h,i,u,j)$ and in $C(h,j,v,i)$. If uv is "not selected" (with respect to a coloring completion) then the two local connectivities yield two components of that color.

The argument in the other direction is a little more subtle. First of all, one should verify that the gadgets enforce some restrictions on any solution for G':

(1) Each Choice module necessarily "outputs" one unsolved color (that is, a color not connected into a single component locally), and thus a Co-Incident Edge Set module $E(v)$ "outputs" $k-1$ colors representing edges incident on v.
(2) Each XOR module forces the "output" stream of unsolved colors to be one, or the other, but not a mixture, of the two input streams of unsolved colors.

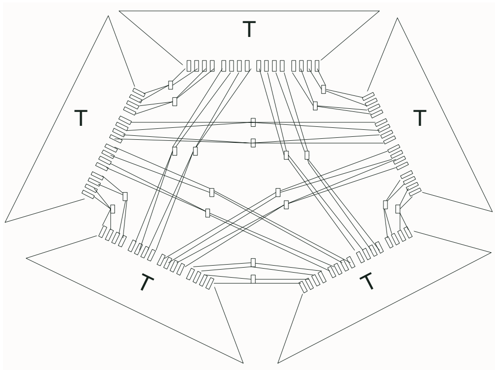

Fig. 6. The modular design of G' for $k = 5$

(3) The unsolved colors that are presented to the central gadget (see Figure 7) can be solved only if these unsolved colors occur in pairs.

The treewidth of G' is easily seen to be $O(k^2)$.

References

1. Arnborg, S., Lagergren, J., Seese, D.: Easy problems for tree-decomposable graphs. J. Algorithms 12, 308–340 (1991)
2. Alon, N., Yuster, R., Zwick, U.: Color-coding. Journal of the ACM 42, 844–856 (1995)
3. Bodlaender, H.L., Fellows, M., Langston, M., Ragan, M.A., Rosamond, F., Weyer, M.: Quadratic kernelization for convex recoloring of trees. Proceedings COCOON 2007, these proceedings (2007)
4. Bar-Yehuda, R., Feldman, I., Rawitz, D.: Improved approximation algorithm for convex recoloring of trees. In: Erlebach, T., Persinao, G. (eds.) WAOA 2005. LNCS, vol. 3879, pp. 55–68. Springer, Heidelberg (2006)
5. Borie, R.B., Parker, R.G., Tovey, C.A.: Automatic generation of linear-time algorithms from predicate calculus descriptions of problems on recursively generated graph families. Algorithmica 7, 555–581 (1992)
6. Bodlaender, H.L., Weyer, M.: Convex anc connected recolourings of trees and graphs. Manuscript (2005)
7. Bu, D., Zhao, Y., Cai, L., Xue, H., Zhu, X., Lu, H., Zhang, J., Sun, S., Ling, L., Zhang, N., Li, G., Chen, R.: Topological structure analysis of the protein-protein interaction network in budding yeast. Nucleic Acids Res. 31(9), 2443–2450 (2003)
8. Chen, J., Chor, B., Fellows, M., Huang, X., Juedes, D., Kanj, I., Xia, G.: Tight lower bounds for certain parameterized NP-hard problems. In: Proceedings of the 19th Annual IEEE Conference on Computational Complexity, pp. 150–160. IEEE Computer Society Press, Los Alamitos (2004)
9. Courcelle, B.: The monadic second-order logic of graphs I: Recognizable sets of finite graphs. Information and Computation 85, 12–75 (1990)
10. Downey, R.G., Fellows, M.R.: Parameterized Complexity. Springer, Heidelberg (1999)

11. Flum, J., Grohe, M.: Parameterized Complexity Theory. Springer, Heidelberg (2006)
12. Fellows, M., Giannopoulos, P., Knauer, C., Paul, C., Rosamond, F., Whitesides, S., Yu, N.: The lawnmower and other problems: applications of MSO logic in geometry, Manuscript (2007)
13. Gramm, J., Nickelsen, A., Tantau, T.: Fixed-parameter algorithms in phylogenetics. Manuscript (2006)
14. Huson, D.H., Bryant, D.: Application of phylogenetic networks in evolutionary studies. Mol. Biol. E 23, 254–267 (2006)
15. Huson, D.H.: SplitsTree: a program for analyzing and visualizing evolutionary data. Bioinfomatics 14, 68–73 (1998)
16. Kelley, B.P., Sharan, R., Karp, R.M., Sittler, T., Root, D.E., Stockwell, B.R., Ideker, T.: Conserved pathways within bacteria and yeast as revealed by global protein network alignment. Proc. Natl. Acad. Sci. USA 100, 11394–11399 (2003)
17. Moran, S., Snir, S.: Convex recolorings of strings and trees: definitions, hardness results and algorithms. To appear in Journal of Computer and System Sciences. In: Dehne, F., López-Ortiz, A., Sack, J.-R. (eds.) WADS 2005. LNCS, vol. 3608, pp. 218–232. Springer, Heidelberg (2005) A preliminary version appeared
18. Moran, S., Snir, S., Sung, W.: Partial convex recolorings of trees and galled networks. Manuscript (2006)
19. Niedermeier, R.: Invitation to Fixed Parameter Algorithms. Oxford University Press, Oxford (2006)
20. Ramadan, E., Tarafdar, A., Pothen, A.: A hypergraph model for the yeast protein complex network. Fourth IEEE International Workshop on High Performance Computational Biology, Santa Fe, NM, April 26, 2004. IEEE Computer Society Press, Los Alamitos (2004)
21. Rual, J.F., Venkatesan, K., Hao, T., Hirozane-Kishikawa, T., Dricot, A., Li, N., Berriz, G.F., Gibbons, F.D., Dreze, M., Ayivi-Guedehoussou, N., Klitgord, N., Simon, C., Boxem, M., Milstein, S., Rosenberg, J., Goldberg, D.S., Zhang, L.V., Wong, S.L., Franklin, G., Li, S., Albala, J.S., Lim, J., Fraughton, C., Llamosas, E., Cevik, S., Bex, C., Lamesch, P., Sikorski, R.S., Vandenhaute, J., Zoghbi, H.Y., Smolyar, A., Bosak, S., Sequerra, R., Doucette-Stamm, L., Cusick, M.E., Hill, D.E., Roth, F.P., Vidal, M.: Nature, 437, 1173–1178 (2005)
22. Schwikowski, B., Uetz, P., Fields, S.: A network of protein-protein interactions in yeast. Nature Biotechnology 18(12), 1257–1261 (2000)
23. Viveshwara, S., Brinda, K.V., Kannan, N.: Protein structure: insights from graph theory. J. Theoretical and Computational Chemistry 1, 187–211 (2002)