New Results on the Complexity of the Max- and Min-Rep Problems

Robert Ganian

Faculty of Informatics,
Masaryk University, Brno, Czech Republic.

SOFSEM 2011

Outline

Max-Rep and Min-Rep

- The *Max-Rep* and *Min-Rep* problems were both introduced by Kortsarz as variants of the Label Cover problem
- Both problems are often used in hardness reductions (Directed Steiner Forest, l-Round Power Dominating Set, Red-Blue Set Cover, Target Set Selection, and many others)
- So, new complexity results here may be used to obtain new results on a wide range of other problems as well

Max-Rep and Min-Rep

- The *Max-Rep* and *Min-Rep* problems were both introduced by Kortsarz as variants of the Label Cover problem
- Both problems are often used in hardness reductions (Directed Steiner Forest, l-Round Power Dominating Set, Red-Blue Set Cover, Target Set Selection, and many others)
- So, new complexity results here may be used to obtain new results on a wide range of other problems as well
- Previous resarch has focused on approximation algorithms, and it turns out both are strongly inapproximable (i.e. inapproximable with ratio $2^{log^{1-e} n}$ for any e)

Problem definitions

Definition (Max-Rep)

Instance: A bipartite graph $G = (A, B, E)$ with $n = |A| = |B| = m \cdot k$ and $2m$ cardinality-k disjoint subsets $A_1 \ldots A_m$, $B_1 \ldots B_m$ such that $A_1 \cup A_2 \cup \cdots \cup A_m = A$ and $B_1 \cup B_2 \cup \cdots \cup B_m = B$.

Objective: Select sets of representatives $A' \subseteq A$, $B' \subseteq B$ such that $|A' \cap A_i| = 1$, $|B' \cap B_i| = 1$ for all $1 \leq i \leq m$ and the subgraph induced by $A' \cup B'$ has the maximum number of edges.

Problem definitions

Definition (Min-Rep)

Instance: A bipartite graph $G = (A, B, E)$ with $n = |A| = |B| = m \cdot k$ and $2m$ cardinality-k disjoint subsets $A_1 \ldots A_m$, $B_1 \ldots B_m$ such that $A_1 \cup A_2 \cup \cdots \cup A_m = A$ and $B_1 \cup B_2 \cup \cdots \cup B_m = B$.

Objective: Select sets $A' \subseteq A$, $B' \subseteq B$ such that each pair of sets A_i, B_j adjacent in G is also adjacent in $G[A' \cup B']$. The goal is to minimize $|A'| + |B'|$.

Parameterized Complexity

Instead of using approximation techniques, we tackle these problems with *parameterized complexity*

- In most cases, it is not necessary to solve problems on general graphs and inputs; some structure is present
- This structure can be characterized by a parameter
- We can then design algorithms which are polynomial for any fixed value of the parameter k: FPT algorithms (time $O(f(k) \cdot poly(n))$) and XP algorithms (time $O(poly(n)^{f(k)})$)

Choosing the right parameter

There are two main approaches to selecting a parameter:

1. Either use the parameter to capture the structure of the graph – i.e. solving problems on graphs of bounded rank-width or tree-width, or
2. obtain the parameter from the problem itself – i.e. finding an independent set when its size is bounded

It turns out that the first approach does **not** help, however the second one does yield some interesting results.

Outline

Bounding the number of sets

It is a trivial observation that Max-Rep may be computed in XP time when the number of sets is bounded – one may simply run through all possibilities of choosing A' and B' in time $(\frac{n}{m})^{2m} \cdot n^2$.
A similar approach also works for Min-Rep, yielding a $O((\frac{n^2}{m^2})^{m^2} \cdot n)$ algorithm.
But what about FPT algorithms?

$W[1]$-hardness I

Theorem

Max-Rep and Min-Rep are both $W[1]$-hard when parameterized by m.

$W[1]$-hardness I

Theorem

Max-Rep and Min-Rep are both $W[1]$-hard when parameterized by m.

Proof (sketch):

- In this case it is possible to use a reduction from Independent Set for both problems
- When asked for an independent set of size d in I, we construct a Max/Min-Rep instance with $6d$ sets $A_1 \dots A_{3d}$ and $B_1 \dots B_{3d}$
- Each set A_i and B_j corresponds to the vertex set in in I

$W[1]$-hardness II

- Next, for $1 \leq i, j \leq d$, vertices between A_i and B_j will be adjacent iff they are **not** adjacent in I.

$W[1]$-hardness II

- Next, for $1 \le i, j \le d$, vertices between A_i and B_j will be adjacent iff they are **not** adjacent in I.
- Notice that if we ensure the following:
  1. Vertices selected in A_i are the same (in I) as those selected in B_i
  2. No two sets $A_z \ne A_x$ have selected the same vertex in G,

  then Max-Rep would return d^2 and Min-Rep would select only one vertex in each set **if and only if** the vertices selected in $A_1 \dots A_d$ form an independent set in I
- the remaining 2d pairs of sets are simply used to ensure that these conditions hold

$W[1]$-hardness III

1. Vertices selected in A_i are the same (in I) as those selected in B_i:
   To ensure that this condition is preserved in the optimal solution, we use A_i–B_{i+d}–A_{i+d}–B_i paths between the same vertices in I. *We reward "bonus points" for playing fair*.
2. No two sets $A_z \neq A_x$ have selected the same vertex in G:
   Here we add an edge between A_i and B_{i+2d} between the same I-vertices, and then edges between $A_{j \neq i}$ and B_{i+2d} between different I-vertices.

$W[1]$-hardness IV

The key observation is that these additions do not change the fact that a theoretically optimal Max-Rep (of size $3d^2 + 3d$) and Min-Rep solution (1 vertex in each set) only occurs if $A_1 \dots A_d$ and $B_1 \dots B_d$ contain an independent set in I.

□

FPT algorithms

- These results suggest that FPT algorithms are unlikely to exist for Max/Min-Rep parameterized solely by m
- However, additionally parameterizing by graph structural parameters does result in FPT algorithms:

FPT algorithms

- These results suggest that FPT algorithms are unlikely to exist for Max/Min-Rep parameterized solely by m
- However, additionally parameterizing by graph structural parameters does result in FPT algorithms:

Theorem

There exists an algorithm computing Max-Rep in FPT time parameterized by the tree-width of the input graph and m.

Theorem

There exists an algorithm computing Min-Rep in FPT time parameterized by the rank-width of the input graph and m.

FPT algorithms II

Theorem

There exists an algorithm computing Max-Rep in FPT time parameterized by the tree-width of the input graph and m.

Theorem

There exists an algorithm computing Min-Rep in FPT time parameterized by the rank-width of the input graph and m.

- Both algorithms result from a *LinEMSOL* expression
- Surprisingly, an MS_1 expression suffices for Min-Rep
- What about just bounding the tree-width and ignoring the m parameter?

Outline

Bounding the size of sets

- Parameterizing by m yielded at least some positive results
- Obviously, parameterizing by k only makes sense for $k \geq 2$
- And the results?

Theorem

Max-Rep is NP-hard to compute when $k \geq 2$, even if G is a forest with maximum path length 2 and maximum degree 3.

Bounding the size of sets II

Theorem

Max-Rep is NP-hard to compute when $k \geq 2$, even if G is a forest with maximum path length 2 and maximum degree 3.

- Max-Rep remains hopeless even on very simple graphs

Max-Rep reduction

- We reduce from the Max-2SAT problem with exactly 3 occurrences of each variable
- Each variable occurs positively either once or twice
- Let every clause c be represented by some $B_c = \{c_1, c_2\}$, with both vertices of degree 1
- Each occurrence of a literal in a clause will be represented by a vertex in A with an edge to the appropriate clause
- Now, if we could ensure that only positive or negative occurrences of each variable are allowed to be selected, then each edge Max-Rep finds corresponds to a satisfied clause

Max-Rep reduction II

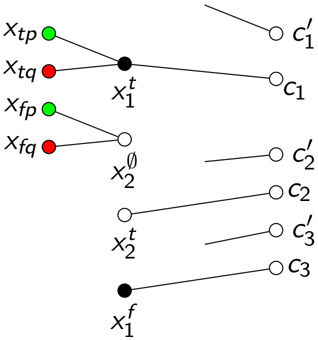


Figure: An illustration of the Max-Rep reduction gadget for variable x

Min-Rep

Theorem

Min-Rep is NP-hard to compute when $k \geq 2$.

- Here the reduction gadgets for Max-Rep and Min-Rep are in fact different

Min-Rep reduction (sketch)

We again reduce from the Max-2SAT problem with exactly 3 occurrences of each variable

Min-Rep reduction (sketch)

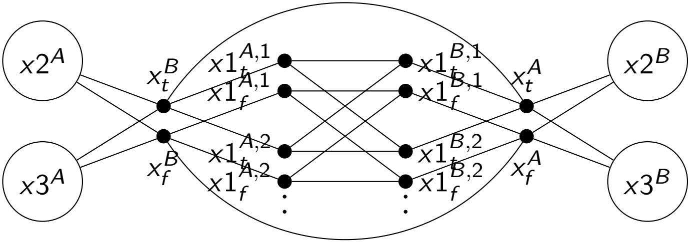
We again reduce from the Max-2SAT problem with exactly 3 occurrences of each variable The problem may also be equivalently stated as follows:

1. Given is a 2SAT instance and the goal is to satisfy all clauses
2. However, you are allowed to select variables which "cheat" and satisfy all clauses they appear in
3. The goal is to minimize the number of cheating variables

Min-Rep reduction (sketch) II

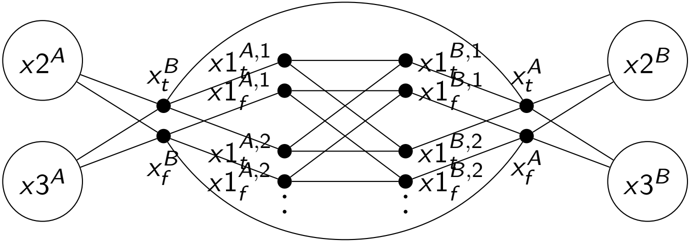
The reduction works as follows:

- A and B are fully symmetric
- For each variable, we create a control set with one vertex representing *true* and the other *false*
- Next, for each occurrence of that variable we create many 2-sets – each vertex representing true and false – and ensure that (without cheating) all the occurrences are either true or false



Min-Rep reduction (sketch) III

- If both vertices in the control set are selected, then that variable is allowed to cheat and the occurrences become independent of each other
- Finally, clauses are added by adding edges between the appropriate occurrences (i.e. between $y2_f$ and $x3_t$)*



Outline

Conclusions

- We have provided several new complexity results for the Min-Rep and Max-Rep problems
- These problems are often used in hardness reductions, and so the obtained results may be transferable to a wide range of other problems
- The article underscores the importance of selecting the correct parameter for tackling individual problems – compare the results obtained when parameterizing by k and by m
- The complexity of Min-Rep remains open when $k = 2$ on graphs of bounded structural parameters

Thank
You
For
Your
Attention!