Turbo charging heuristics: adjusting the parameters for optimum performance. (Talk 2)

Faisal N. Abu-Khzam\textsuperscript{1,2} \quad Shaowei Cai\textsuperscript{3} \quad Judith Egan\textsuperscript{1} \quad Peter Shaw\textsuperscript{4} \quad Kai Wang\textsuperscript{1}

Charles Darwin University AU

Lebanese American University, Beirut

Massey University, Manawatu

Chinese Academy of Sciences, Beijing

July 2018
Abstract

Turbo-charging is a recent algorithmic technique for hard problems that employs an FPT subroutine as part of a heuristic. We demonstrate the effectiveness of this technique and develop the turbo-charging idea further. In this talk we will explore how the performance can be improved through adjusting the parameters and moment-of-regret function.

We implement both the initially proposed “turbo-greedy” algorithm of Downey et al. and a new hybrid heuristic for the $W[2]$-hard DOMINATING SET problem and evaluated their performance for a range of parameters and datasets. Our algorithm often produced results that were either exact or better than all the other available heuristic algorithms. The results vary depending on the parameter, with the best results obtained for larger values of $k$ and $r$. 

Faisal N. Abu-Khzam, Shaowei Cai, Judith Egan, Peter Shaw (Charles Darwin University AU, Lebanese American University, Beirut, Massey University, Manawatu, Chinese Academy of Sciences, Beijing)

Turbo charging heuristics: adjusting the parameters for optimum performance. (Talk 2)

July 2018
Overview

1. Motivation
2. Dynamic FPT Heuristics
3. Greedy FPT
4. Greedy DDS Algorithm
5. Hybrid DDS Algorithm
6. Varying Moment of regret
7. Experimental results
**Motivation**

Figure: The Dynamic FPT formulations of some $W$-hard problems are in FPT.

- **FPT**
  - $T(n, k) = 2^{O(k)} \cdot n^c$

- **W Hierarchy**
  - $FPT \subseteq W[1] \subseteq W[2] \subseteq \cdots XP$

- Ideally, Turbo Charging provides a $\alpha^f(r,k)n$ heuristics with an exchange neighborhood of $r$. 
**Definition**

**Dominating Set (DS)**

*Instance*: A graph $G = (V, E)$

*Parameter*: $k$

*Question*: Does $G$ have a dominating set $S \subseteq V$, $|S| \leq k$, such that every vertex in $V \setminus S$ is adjacent to a vertex in $S$?

---

**Figure**: Example of Dominating Set (DS)
Terminology

- **open/closed neighborhood**
  - $N_G(S) = \bigcup_{v \in S} N(v)$
  - $N_G[S] = N_G(S) \cup S$

- **Measure**
  - $utility(v) =$ the number of non-dominated neighbors of the vertex $v$
  - $vote(v) = utility(v) + \frac{1}{\sum_{u \in N(v)} utility(u)}$ [10]

- **Edit Distance**
  - $d_e(G, G') = |(E(G) \setminus E(G')) \cup (E(G') \setminus E(G))|$
  - $d_v(D, D') = |(D \setminus D') \cup (D' \setminus D)|$
Greedy Heuristics

- **Greedy Chvátal** [3]
- **Greedy Vote** [10]
- **Greedy Vote GRASP** [10]
Dynamic FPT Heuristics

- FPT Turbo I [4]
Definition

Dynamic Dominating Set (DDS)

Instance: A graph $G$ and a graph $G'$ with $d_e(G, G') \leq k$, a dominating set $S \subseteq V(G)$; a positive integer $r$.

Parameter: $(k, r)$

Question: Does there exist a set of vertices $S' \subseteq V(G') = V(G)$ with $d_v(S, S') \leq r$ and with $S'$ being a dominating set of $G'$?

Definition

Greedy Improvement of Dominating Set (Greedy-DS) [4]

Instance: A graph $G$; a list $L$ of the vertices of $G$ ordered from highest degree to lowest degree,

$L = (v_1, \ldots, v_l, \ldots, v_{l+k} = v, \ldots, v_n)$; a set of vertices $D \subseteq V(G)$ that dominates the set $V' = \{v_1, \ldots, v_{l+k}\}$; a partition $D = D_1 \cup D_2$ where $D_1$ dominates the set of vertices $\{v_1, \ldots, v_l\}$ and $|D_2| \leq k$

Parameter: $k$

Question: Is there a set of vertices $D' \subseteq V(G)$ such that $D'$ dominates the vertices of $V' \cup D$, with $D_1 \subseteq D'$ and $|D'| < |D|$?
Greedy Improvement of Dominating Set (Greedy-DS)

Figure: Illustration of Greedy-DS
Greedy DDS

Require: Two graphs $G$ and $G'$, parameters $k$ and $r$, and a dominating set solution $S$ for $G$. 
1: Divide graph $G'$ into three parts: $S$, $B$, and $C$, where 
2: $C \leftarrow V(G') \setminus N_{G'}[S]$ 
3: $B \leftarrow N_{G'}[S] \setminus S$; 
4: Apply the reduction rules [4] on $G'$ to get a reduced instance $G^*$ with $B^*$ and $C$; $\triangleright C$ is a vertex cover for $G^*$ and in turn $B^*$ is an independent set for $G^*$ 
5: if $B^*$ and $C$ are the same as history record, return $S$; 
6: Traverse vertices in $B^*$ and $C$ to get a vertex set $P$ which presents different neighbor types of $C$; $\triangleright |P| \leq \text{slant}2^k$ because $|C| \leq \text{slant}k$ 
7: apply rule (R???) on the neighbor types of $C$ so as to remove impossible vertices in $P$; 
8: findSolution $\leftarrow$ false; 
9: do 
10: Choose $r$ vertices from $P$ to construct a set $D$; 
11: if $D$ dominates $C$ then 
12: findSolution $\leftarrow$ true; 
13: goto Line 19 
14: end if 
15: while (all combinations of $r$ vertices are not tried) 
16: if findSolution = false then 
17: $D \leftarrow C$; 
18: end if 
19: $S' \leftarrow S \cup D$ 
20: Return $S'$ as the dominating set solution of of graph $G'$;
Developing a new hybrid algorithm

A new FPT Turbo Hybrid algorithm.

- Connected Components;
- Applying reduction rules;
- Using alternative measure(s) for selecting solution elements;
- Using dynamic re-ordering of vertex list on the measures;
- Checking whether the solution(s) obtained are minimal;
- Applying an appropriate LS heuristic;
- Adding a heuristic guarantee;
- Specifying an appropriate Moment_of_Regret function
Input: a graph $G = (V, E)$; parameters $k$ and $r_{upper}$ such that $r_{upper} = k - 1$ (initially)

2: **Rank** a list $L$ of vertices in $G$ from lowest to highest utility;

3: **Get** the vertex $v_0$ of the lowest utility;

4: **Get** the highest utility vertex $u_0 \in N_G[v_0]$;

5: $S_0 \leftarrow \{u_0\}$;

6: **Initialize** the graph $G_0$ with $\{u_0, v_0\}$ and the edge between $u_0$ and $v_0$;

7: $i \leftarrow 0$;

8: **do**

9: $i \leftarrow i + 1$;

10: **Rank** the list $L$ of the vertices (Vote or utility);

11: if $v_i$ is dominated by $S_{i-1}$ then

12: $G_i \leftarrow G_{i-1} \cup \{v_i\}$;

13: $S_i \leftarrow S_{i-1}$;

14: else

15: **Get** the highest utility vertex $u_i \in N_G[v_i]$;

16: $S_i \leftarrow S_{i-1} \cup \{u_i\}$;

17: **Construct** $G_i$ from $G_{i-1}$ with $\{v_i, u_i\}$ and incident edges in $G$;

18: if $\text{is_moment_of_regret}(G_i, u_i, S_i, S_{i-k})$ then

19: $r \leftarrow \min(r_{upper}, |S_i| - |S_{i-k}| - 1)$;

20: $\hat{G}_i \leftarrow \text{a virtually constructed graph from } G_i \text{ by adding } \leq 2k \text{ edges between } S_{i-k} \text{ and } V(G_i) \setminus N_{G_i}[S_{i-k}]$;

21: $S_i' \leftarrow \text{DDS FPT } (G = \hat{G}_i, G' = G_i, S = S_{i-k}, k = |V(G_i) \setminus V(G_{i-k})|, r = r)$;

22: $S_i \leftarrow \min(S_i, S_i')$ \hspace{1cm} **Get the minimum size set**

23: end if

24: end if

25: **while** (Not all the vertices are dominated);

26: **Return** the final $S_i$ as the dominating set solution for $G$;
\{v_0, \ldots, v_l\} \quad S_{i-k}

(a) \quad G_{i-k}

\{v_{l+1}, \ldots, v_{l+k}\} \quad S_i \setminus S_{i-k}

(b) \quad \hat{G}_i

Figure: \quad G_i \longrightarrow \hat{G}_i

- any dashed-edges are removed by reduction rule so don’t effect \(k\)
A new Moment-of-regret function

- In the new `is_moment_of_regret` we allow
  \[(|S_i| - |S_{i-k}|) \geq MOR_{threshold}.\]
- Each time the moment-of-regret happens the parameter value of \( r \) can vary
- So \( r \) for each time should be the minimum among
  1. \((|S_i| - |S_{i-k}| - 1)\);
  2. at least one less than \((|S_{i}^{t} - |S_{i-k}| - 1)\), \( S_{i}^{t} \) is the solution size of heuristic guarantee;
  3. the input argument \( r_{upper} \). (New input value)
All algorithms presented in this paper are implemented in the Java programming language. The experiments are run on a computer of the OSX Yosemite operating system with a CPU of 3.5 GHz, 6-Core Intel Xeon E5, and 32 GB memory. Exact solutions shown using Fomin et al. [5], implemented using the hybrid method of Abu-Khzam et al. [2] given

- KONECTS [1, 6]
- KONECTS (Massive graphs $|V| > 4900$)
- DIMACS (Complement) [8]
- BSHOLIB [?]
**Table**: Algorithm performance on the KONECT data sets (Time in sec).

- FPT Turbo Hybrid out-perform others
- FPT Turbo I better size/time ratio
Performance on the large KONECT Data Sets

| Data Set         | Name              | |V| | E| | Size | Time | Size | Time |
|------------------|-------------------|---|---|---|---|---|---|---|---|
|                  | Power Grid        |   | 4941 | 6594 | 1588 | 3.7 | 1499 | 37.6 |
|                  | Pretty Good Privacy |   | 10680 | 24316 | 2862 | 129 | 2732 | 287 |
|                  | arXiv astro-ph    |   | 18771 | 198050 | 2509 | 628 | 2456 | 555 |
|                  | CAIDA             |   | 26475 | 53381 | 2422 | 447 | 2406 | 2411 |

Table: A comparison of the algorithm performance on the large KONECT data sets (Time in sec).

- Due to the size of these graphs, the exact algorithm and GRASP Local Search heuristic were not able to process the graphs.
- NOTE FPT Turbo runs faster than Greedy for some sets.
Solution size were measured on the Power Grid instance. chosen as it provided a large enough solution size to give a range of results.

823 results were recorded

\[ 1 \geq r_{upper} \geq (k - 1), \quad 3 \geq k \geq 15 \quad \text{and} \quad 1 \geq \text{threshold} \geq k \]

These results have been classified using a Recursive partitioning [11]

Best solution size were obtained with \( r_{upper} \geq 5 \) threshold \( t \) between \( 5 \geq t \geq 9 \) and \( k = 12 \).
Effect of $r_{upper}$, k, and Moment of Regret Trigger (threshold) on solution size

- $r_{upper} \geq 4.5$
- $r_{upper} < 4.5$
- $r_{upper} \geq 7.5$
- $r_{upper} < 7.5$
- $r_{upper} \geq 2.5$
- $r_{upper} < 2.5$

- $k \geq 12$
- $k < 12$
- $k \geq 9.5$
- $k < 9.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$
- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$
- $\text{threshold} \geq 4.5$
- $\text{threshold} < 4.5$

- $\text{threshold} \geq 7.5$
- $\text{threshold} < 7.5$
- $\text{threshold} \geq 2.5$
- $\text{threshold} < 2.5$

- $\text{threshold} \geq 14$
- $\text{threshold} < 14$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 14$
- $\text{threshold} < 14$

- $\text{threshold} \geq 9.5$
- $\text{threshold} < 9.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$

- $\text{threshold} \geq 12$
- $\text{threshold} < 12$

- $\text{threshold} \geq 6.5$
- $\text{threshold} < 6.5$
Effect of parameters on Time

- Of the 823 tests, only 43 obtained the minimum solution of 1499.
- this result was notably less than the greedy Chvátal solution of 1588.
- It seem interesting to then consider the execution times of these optimal solutions.
Figure: Execution times for moment of regret (threshold) \( r_{upper} \), and \( k \) resulting in optimum solutions (KONECT–Power Grid Network). (Generated using the Rctree package [7])
Conclusion and Future Work

- Happy that the FPT procedure can have useful results
- For best results set $r > 5$ and use a none-trivial moment of regret function
- Obviously ordering effects the result
- New LS heuristics should be available soon to compare the results
- Seems to give best results on scale-free graphs rather than graphs where greedy heuristics gave an optimum solution.
References

U. rovira i virgili network dataset – KONECT, April 2015.


## Performance on the DIMACS data set

<table>
<thead>
<tr>
<th>Data Set</th>
<th>V</th>
<th>E</th>
<th>Opt</th>
<th>Greedy Chvátal</th>
<th>Greedy Vote GRASP</th>
<th>FPT Turbo I</th>
<th>FPT Turbo Hybrid</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td></td>
<td></td>
<td></td>
<td>Size</td>
<td>Size</td>
<td>Time</td>
<td>Size</td>
</tr>
<tr>
<td>C1000.9</td>
<td>1000</td>
<td>450079</td>
<td>2</td>
<td>3</td>
<td>0.83</td>
<td>3</td>
<td>2.88</td>
</tr>
<tr>
<td>C125.9</td>
<td>125</td>
<td>6963</td>
<td>2</td>
<td>2</td>
<td>0</td>
<td>2</td>
<td>0.01</td>
</tr>
<tr>
<td>C2000.5</td>
<td>2000</td>
<td>999836</td>
<td>NA</td>
<td>7</td>
<td>5.56</td>
<td>7</td>
<td>42.5</td>
</tr>
<tr>
<td>C2000.9</td>
<td>2000</td>
<td>1799532</td>
<td>NA</td>
<td>3</td>
<td>3.83</td>
<td>3</td>
<td>13.4</td>
</tr>
<tr>
<td>C250.9</td>
<td>250</td>
<td>27984</td>
<td>2</td>
<td>2</td>
<td>0.02</td>
<td>2</td>
<td>0.06</td>
</tr>
<tr>
<td>C4000.5</td>
<td>4000</td>
<td>4000268</td>
<td>NA</td>
<td>7</td>
<td>25.9</td>
<td>8</td>
<td>234</td>
</tr>
<tr>
<td>C500.9</td>
<td>500</td>
<td>112332</td>
<td>2</td>
<td>2</td>
<td>0.11</td>
<td>2</td>
<td>0.31</td>
</tr>
<tr>
<td>DSJC1000.5</td>
<td>1000</td>
<td>249826</td>
<td>NA</td>
<td>6</td>
<td>1.02</td>
<td>6</td>
<td>6.95</td>
</tr>
<tr>
<td>DSJC500.5</td>
<td>500</td>
<td>62624</td>
<td>NA</td>
<td>5</td>
<td>0.18</td>
<td>5</td>
<td>1.01</td>
</tr>
<tr>
<td>MANN_a27</td>
<td>378</td>
<td>70551</td>
<td>2</td>
<td>2</td>
<td>0.06</td>
<td>2</td>
<td>0.16</td>
</tr>
<tr>
<td>MANN_a81</td>
<td>3321</td>
<td>5506380</td>
<td>NA</td>
<td>2</td>
<td>11.5</td>
<td>2</td>
<td>25.4</td>
</tr>
<tr>
<td>brock200_2</td>
<td>200</td>
<td>9876</td>
<td>NA</td>
<td>5</td>
<td>0.02</td>
<td>4</td>
<td>0.09</td>
</tr>
<tr>
<td>brock200_4</td>
<td>200</td>
<td>13089</td>
<td>3</td>
<td>3</td>
<td>0.02</td>
<td>4</td>
<td>0.1</td>
</tr>
<tr>
<td>brock400_2</td>
<td>400</td>
<td>59786</td>
<td>3</td>
<td>3</td>
<td>0.09</td>
<td>3</td>
<td>0.31</td>
</tr>
<tr>
<td>brock400_4</td>
<td>400</td>
<td>59765</td>
<td>3</td>
<td>3</td>
<td>0.09</td>
<td>3</td>
<td>0.36</td>
</tr>
<tr>
<td>brock800_2</td>
<td>800</td>
<td>208166</td>
<td>4</td>
<td>4</td>
<td>0.53</td>
<td>4</td>
<td>2.38</td>
</tr>
<tr>
<td>brock800_4</td>
<td>800</td>
<td>207643</td>
<td>4</td>
<td>4</td>
<td>0.51</td>
<td>4</td>
<td>2.46</td>
</tr>
<tr>
<td>gen200_p0.9_44</td>
<td>200</td>
<td>17910</td>
<td>2</td>
<td>2</td>
<td>0.01</td>
<td>2</td>
<td>0.04</td>
</tr>
<tr>
<td>gen200_p0.9_55</td>
<td>200</td>
<td>17910</td>
<td>2</td>
<td>2</td>
<td>0.01</td>
<td>2</td>
<td>0.04</td>
</tr>
<tr>
<td>gen400_p0.9_55</td>
<td>400</td>
<td>71820</td>
<td>2</td>
<td>2</td>
<td>0.07</td>
<td>2</td>
<td>0.19</td>
</tr>
<tr>
<td>gen400_p0.9_65</td>
<td>400</td>
<td>71820</td>
<td>2</td>
<td>2</td>
<td>0.07</td>
<td>2</td>
<td>0.19</td>
</tr>
<tr>
<td>gen400_p0.9_75</td>
<td>400</td>
<td>71820</td>
<td>2</td>
<td>2</td>
<td>0.07</td>
<td>2</td>
<td>0.19</td>
</tr>
<tr>
<td>hamming10-4</td>
<td>1024</td>
<td>434176</td>
<td>2</td>
<td>2</td>
<td>0.53</td>
<td>2</td>
<td>1.47</td>
</tr>
<tr>
<td>hamming8-4</td>
<td>256</td>
<td>20864</td>
<td>2</td>
<td>2</td>
<td>0.02</td>
<td>2</td>
<td>0.05</td>
</tr>
<tr>
<td>keller4</td>
<td>171</td>
<td>9435</td>
<td>2</td>
<td>2</td>
<td>0.01</td>
<td>2</td>
<td>0.02</td>
</tr>
<tr>
<td>keller5</td>
<td>776</td>
<td>225990</td>
<td>2</td>
<td>2</td>
<td>0.25</td>
<td>2</td>
<td>0.7</td>
</tr>
<tr>
<td>keller6</td>
<td>3361</td>
<td>4619898</td>
<td>2</td>
<td>2</td>
<td>8.47</td>
<td>2</td>
<td>26.7</td>
</tr>
</tbody>
</table>
### Performance on the DIMACS (Compliment) data set

<p>| DataSet       | $|V|$ | $|E|$ | Opt | Greedy Chvátal | Greedy Vote | FPT Turbo I | FPT Turbo Hybrid II |
|---------------|-----|-----|-----|----------------|-------------|-------------|---------------------|
|               |     |     |     | Size           | Size        | Size        | Size                |
|               | V   | E   | Time| Size           | Time        | Size        | Time                |
| brock200_2    | 200 | 10024 | 4   | 0.007          | 4           | 0.044       | 4                  | 0.011               |
| brock200_4    | 200 | 6811  | 7   | 0.003          | 6           | 0.025       | 6                  | 0.003               |
| brock400_2    | 400 | 20014 | 10  | 0.015          | 9           | 0.067       | 10                 | 0.012               |
| brock400_4    | 400 | 20035 | 10  | 0.005          | 10          | 0.032       | 9                  | 0.006               |
| brock800_2    | 800 | 111434| 8   | 0.007          | 8           | 0.095       | 8                  | 0.010               |
| brock800_4    | 800 | 111957| 8   | 0.006          | 8           | 0.105       | 8                  | 0.030               |
| C1000.9       | 1000| 49421 | 25  | 0.021          | 24          | 0.351       | 31                 | 0.037               |
| C125.9        | 125 | 787   | 14  | 0.001          | 14          | 0.007       | 14                 | 0.001               |
| C2000.5       | 2000| 999164| 7   | 0.026          | 7           | 0.497       | 7                  | 0.215               |
| C250.9        | 250 | 3141  | 18  | 0.001          | 17          | 0.014       | 19                 | 0.002               |
| C4000.5       | 4000| 3997732| 8   | 0.089          | 8           | 3.073       | 8                  | 2.604               |
| C500.9        | 500 | 12418 | 21  | 0.003          | 20          | 0.137       | 29                 | 0.007               |
| DSJC1000.5    | 1000| 249674| 6   | 0.005          | 6           | 0.115       | 6                  | 0.026               |
| DSJC500.5     | 500 | 62126 | 6   | 0.002          | 5           | 0.026       | 6                  | 0.006               |
| gen200_p0.9_55| 200 | 1990  | 17  | 0.001          | 15          | 0.011       | 17                 | 0.002               |
| gen400_p0.9_55| 400 | 7980  | 19  | 0.002          | 19          | 0.042       | 20                 | 0.006               |
| gen400_p0.9_65| 400 | 7980  | 20  | 0.002          | 19          | 0.043       | 24                 | 0.006               |
| gen400_p0.9_75| 400 | 7980  | 21  | 0.003          | 19          | 0.038       | 27                 | 0.007               |
| hamming10-4   | 1024| 89600 | 15  | 0.008          | 14          | 0.289       | 15                 | 0.040               |
| hamming8-4    | 256 | 11776 | 4   | 0.000          | 4           | 0.007       | 4                  | 0.002               |
| keller4       | 171 | 5100  | 6   | 0.000          | 6           | 0.002       | 6                  | 0.001               |
| keller5       | 776 | 74710 | 11  | 0.004          | 11          | 0.119       | 11                 | 0.018               |
| keller6       | 3361| 1026582| 18  | 0.081         | 18          | 3.749       | 18                 | 0.759               |
| MANN_a27      | 378 | 702   | 27  | 0.002          | 27          | 0.073       | 27                 | 0.002               |
| MANN_a45      | 1035| 1980  | 45  | 0.017          | 45          | 0.770       | 45                 | 0.016               |
| MANN_a81      | 3321| 6480  | 81  | 0.190          | 81          | 8.920       | 81                 | 0.354               |
| p_hat1500-1   | 1500| 839327| 3   | 0.007          | 3           | 0.279       | 3                  | 0.009               |
| p_hat1500-2   | 1500| 555290| 5   | 0.009          | 5           | 0.172       | 5                  | 0.012               |
| p_hat1500-3   | 1500| 277096| 11  | 0.013          | 11          | 0.288       | 11                 | 0.015               |</p>
<table>
<thead>
<tr>
<th>Name</th>
<th>V</th>
<th>E</th>
<th>Size</th>
<th>Time</th>
<th>Size</th>
<th>Time</th>
<th>Size</th>
<th>Time</th>
<th>Size</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>frb30-15-1</td>
<td>450</td>
<td>17827</td>
<td>12</td>
<td>0.16</td>
<td>12</td>
<td>1.62</td>
<td>12</td>
<td>0.3</td>
<td>12</td>
<td>19.3</td>
</tr>
<tr>
<td>frb30-15-2</td>
<td>450</td>
<td>17874</td>
<td>12</td>
<td>0.13</td>
<td>12</td>
<td>1.56</td>
<td>13</td>
<td>0.17</td>
<td>12</td>
<td>18.9</td>
</tr>
<tr>
<td>frb30-15-3</td>
<td>450</td>
<td>1809</td>
<td>12</td>
<td>0.12</td>
<td>12</td>
<td>1.65</td>
<td>13</td>
<td>0.16</td>
<td>12</td>
<td>18.4</td>
</tr>
<tr>
<td>frb30-15-4</td>
<td>450</td>
<td>17831</td>
<td>13</td>
<td>0.14</td>
<td>12</td>
<td>1.62</td>
<td>13</td>
<td>0.13</td>
<td>12</td>
<td>19.6</td>
</tr>
<tr>
<td>frb30-15-5</td>
<td>450</td>
<td>17794</td>
<td>13</td>
<td>0.13</td>
<td>12</td>
<td>1.56</td>
<td>13</td>
<td>0.13</td>
<td>12</td>
<td>19.5</td>
</tr>
<tr>
<td>frb35-17-1</td>
<td>595</td>
<td>27856</td>
<td>14</td>
<td>0.25</td>
<td>14</td>
<td>3.52</td>
<td>15</td>
<td>0.21</td>
<td>14</td>
<td>33.7</td>
</tr>
<tr>
<td>frb35-17-2</td>
<td>595</td>
<td>27847</td>
<td>14</td>
<td>0.23</td>
<td>14</td>
<td>3.61</td>
<td>15</td>
<td>0.2</td>
<td>14</td>
<td>40.7</td>
</tr>
<tr>
<td>frb35-17-3</td>
<td>595</td>
<td>27931</td>
<td>15</td>
<td>0.27</td>
<td>14</td>
<td>3.92</td>
<td>16</td>
<td>0.23</td>
<td>14</td>
<td>35.9</td>
</tr>
<tr>
<td>frb35-17-4</td>
<td>595</td>
<td>27842</td>
<td>17</td>
<td>0.28</td>
<td>15</td>
<td>4.29</td>
<td>16</td>
<td>0.2</td>
<td>15</td>
<td>33.5</td>
</tr>
<tr>
<td>frb35-17-5</td>
<td>595</td>
<td>28143</td>
<td>14</td>
<td>0.25</td>
<td>15</td>
<td>4.24</td>
<td>15</td>
<td>0.2</td>
<td>13</td>
<td>45.5</td>
</tr>
<tr>
<td>frb40-19-1</td>
<td>760</td>
<td>41314</td>
<td>16</td>
<td>0.42</td>
<td>16</td>
<td>7.95</td>
<td>17</td>
<td>0.33</td>
<td>16</td>
<td>62.6</td>
</tr>
<tr>
<td>frb40-19-2</td>
<td>760</td>
<td>41263</td>
<td>17</td>
<td>0.46</td>
<td>16</td>
<td>7.43</td>
<td>17</td>
<td>0.33</td>
<td>16</td>
<td>62.7</td>
</tr>
<tr>
<td>frb40-19-3</td>
<td>760</td>
<td>41095</td>
<td>17</td>
<td>0.46</td>
<td>16</td>
<td>7.47</td>
<td>18</td>
<td>0.33</td>
<td>16</td>
<td>67.5</td>
</tr>
<tr>
<td>frb40-19-4</td>
<td>760</td>
<td>41605</td>
<td>17</td>
<td>0.46</td>
<td>16</td>
<td>7.64</td>
<td>17</td>
<td>0.32</td>
<td>16</td>
<td>65</td>
</tr>
<tr>
<td>frb40-19-5</td>
<td>760</td>
<td>41619</td>
<td>16</td>
<td>0.45</td>
<td>16</td>
<td>7.64</td>
<td>17</td>
<td>0.3</td>
<td>16</td>
<td>64.8</td>
</tr>
<tr>
<td>frb45-21-1</td>
<td>945</td>
<td>59186</td>
<td>18</td>
<td>0.72</td>
<td>18</td>
<td>13.7</td>
<td>19</td>
<td>0.44</td>
<td>18</td>
<td>105</td>
</tr>
<tr>
<td>frb45-21-2</td>
<td>945</td>
<td>58624</td>
<td>19</td>
<td>0.78</td>
<td>19</td>
<td>15.2</td>
<td>20</td>
<td>0.43</td>
<td>19</td>
<td>118</td>
</tr>
<tr>
<td>frb45-21-3</td>
<td>945</td>
<td>58245</td>
<td>18</td>
<td>0.72</td>
<td>18</td>
<td>13.8</td>
<td>19</td>
<td>0.48</td>
<td>18</td>
<td>116</td>
</tr>
<tr>
<td>frb45-21-4</td>
<td>945</td>
<td>58549</td>
<td>18</td>
<td>0.69</td>
<td>18</td>
<td>13.8</td>
<td>19</td>
<td>0.47</td>
<td>18</td>
<td>105</td>
</tr>
<tr>
<td>frb45-21-5</td>
<td>945</td>
<td>58579</td>
<td>19</td>
<td>0.71</td>
<td>18</td>
<td>14.8</td>
<td>20</td>
<td>0.44</td>
<td>19</td>
<td>119</td>
</tr>
</tbody>
</table>