AUSTRALASIAN JOURNAL OF COMBINATORICS
Volume **45** (2009), Pages 175–188

The vertex and edge graph reconstruction numbers of small graphs

David Rivshin Stanisław P. Radziszowski

Department of Computer Science
Rochester Institute of Technology
Rochester, NY 14623
U.S.A.
{dfr2936,spr}@cs.rit.edu

Abstract

First posed in 1942 by Kelly and Ulam, the *Graph Reconstruction Conjecture* is one of the major open problem in graph theory. While the Graph Reconstruction Conjecture remains open, it has spawned a number of related questions. In the classical vertex graph reconstruction number problem a vertex is deleted in every possible way from a graph G, and then it can be asked how many (both minimum and maximum) of these subgraphs are required to reconstruct G up to isomorphism. Similar questions can also be posed for the less studied case of edge deletion. For graphs in certain classes there are known formulas to quickly determine reconstruction numbers. However, for the vast majority of graphs the computation devolves to brute force exhaustive search.

Previous computer searches have found the 1-vertex-deletion reconstruction numbers of all graphs of up to 10 vertices. In this paper computed values of 1-vertex-deletion and 1-edge-deletion reconstruction numbers for all graphs on up to 11 vertices are reported. Several examples of graphs with high reconstruction numbers are also presented. This was made possible by an improved algorithm which enabled significant reduction in computation time.

1 Introduction

Traditional graph notation (as in [5, 3, 12]) is primarily used in this paper. In all cases graphs are assumed to be simple, undirected, and finite. Furthermore, graphs are considered to be unlabeled, and therefore isomorphic graphs are not distinguished. In the case of common graphs such as cliques (K_n), bipartite cliques ($K_{r,s}$), paths (P_n), and cycles (C_n), the subscript indicates the number of vertices. Where more complex graphs need to be labeled, the *graph6* notation (as implemented in Brendan McKay's *nauty* package [14]) is used. This notation uses printable ASCII characters to encode the adjacency matrix of the graph in a compact form. As the adjacency matrix of a graph, and therefore the graph6 representation, depends on the vertex labeling, the default canonical labeling from *nauty* is used. For instance, the graph $2K_2$ would be written as `C`` in graph6 notation.

In 1942 Kelly and Ulam proposed the Graph Reconstruction Conjecture, and it has remained an important open problem to this day.

Definition 1 ([11])**.** *Deck*(G) is the multiset of graphs that results from deleting one vertex in every possible way from the graph G. When a vertex is removed, all edges incident to that vertex are also removed. The elements of a *Deck* are customarily referred to as *Card*s.

Graph Reconstruction Conjecture (Kelly and Ulam, 1942 [9, 11])**.** Any simple finite undirected graph G on 3 or more vertices can be uniquely identified (up to isomorphism) by *Deck*(G).

There are no known counter-examples to this conjecture, and it is widely believed to be true [1]. For some classes of graphs the conjecture has been proven to hold; specifically disconnected graphs, regular graphs, trees, and maximal planar graphs [18, 2, 19, 1]. Through exhaustive computer search it has previously been shown that all graphs of between 3 and 11 vertices [13, 16], and certain classes of graphs of up to 16 vertices [13], are reconstructible.

From the original Graph Reconstruction Conjecture, there have arisen many new related problems, such as the Edge Reconstruction Conjecture [12].

Definition 2 ([11])**.** $\mathcal{E}$*Deck*(G) is the multiset of graphs that results from deleting one edge in every possible way from the graph G.

Edge Reconstruction Conjecture (Harary, 1964 [7, 11])**.** Any simple finite undirected graph with 4 or more edges can be uniquely identified (up to isomorphism) by $\mathcal{E}$*Deck*(G).

A graph G is said to be vertex-reconstructible (edge-reconstructible) if it can be uniquely identified (up to isomorphism) from *Deck*(G) ($\mathcal{E}$*Deck*(G)). More recently the question "if a graph is reconstructible, how many of its subgraphs are required to reconstruct it?" has been asked. This takes two forms, the *existential (*or *ally) reconstruction number* ($\exists rn$), and the *universal (*or *adversarial) reconstruction number* ($\forall rn$).

Definition 3 ([8, 20, 11, 1])**.**

- The *existential vertex-reconstruction number* ($\exists vrn$) of a graph G is the cardinality of the smallest $\mathcal{S} \subseteq$ *Deck*(G) that reconstruct G.
- The *universal vertex-reconstruction number* ($\forall vrn$) of a graph G is the smallest number such that all $\mathcal{S} \subseteq$ *Deck*(G) of that cardinality reconstruct G.
- The *existential edge-reconstruction number* ($\exists ern$) of a graph G is the cardinality of the smallest $\mathcal{S} \subseteq \mathcal{E}$*Deck*$(G)$ that reconstruct G.

- The *universal edge-reconstruction number* ($\forall ern$) of a graph G is the smallest number such that all $\mathcal{S} \subseteq \mathcal{E}Deck(G)$ of that cardinality reconstruct G.

If a graph G is not vertex-reconstructible (edge-reconstructible), then we let $\exists vrn(G) = \forall vrn(G) = \infty$ ($\exists ern(G) = \forall ern(G) = \infty$).

In the following, if neither vertex nor edge reconstruction is specifically stated, then the statement applies to both forms. When neither existential nor universal qualifier is given to reconstruction numbers, then the statement applies to both. This differs from convention where vertex reconstruction and existential reconstruction number are assumed if not otherwise specified.

While there is no known direct way to compute the reconstruction number of a general graph, there are various properties that are known. For example, it has been shown by Bollobás that $\exists vrn(G) = \forall vrn(G) = 3$ for almost all graphs [4]. Similarly, Lauri has shown that $\exists ern(G) = \forall ern(G) = 2$ for almost all graphs [10] (see also [1]). There are also a number of classes of graphs which are known to have large (> 3) $\exists vrn$, some of which were recently discovered as a result of computations similar to those described in this paper [18, 17, 16].

2 Vertex Reconstruction Numbers

Table 1 shows the number of graphs with a given $\exists vrn$ and $\forall vrn$ for all graphs with between 3 and 11 vertices, inclusive. The previous search done by McMullen computed these same values for graphs of up to order 10 [16, 15].

		graph order								
		3	4	5	6	7	8	9	10	11
unique graphs		4	11	34	156	1044	12346	274668	12005168	1018997864
$\exists vrn$	3	4	8	34	150	1044	12334	274666	12005156	1018997864
	4		3		4		8		6	
	5				2		2	2	4	
	6						2			
	7								2	
$\forall vrn$	3	4	2	7	8	16	266	45186	6054148	815604300
	4		9	19	56	496	8208	199247	5637886	199382868
	5			8	90	520	3584	28781	301530	3922130
	6				2	12	284	1434	10686	83730
	7						4	20	914	4824
	8								4	12

Table 1: Counts of vertex-reconstruction numbers by number of vertices

It should be noted that the new results on 11 vertices do not show any graphs with high $\exists vrn > 3$. All known constructions for graphs with high $\exists vrn$ require a non-prime number of vertices [18, 17, 16], so this is not unexpected. All graphs with $\exists vrn > 3$, labeled by their graph6 encoding, are listed in Table 2. The graph6 encoding can be directly translated into an adjacency matrix representation using the *nauty* package [14].

It can be seen from Tables 1 and 2 that no graphs where $4 \leq |V(G)| \leq 11$ have both a high $\exists vrn$ and the maximal $\forall vrn$. Those graphs with maximal $\forall vrn$ for each order are listed in Table 3; interestingly all these graphs have $\exists vrn = 3$. The 6 graphs which, along with their complements, have the maximal $\forall vrn(G) = 8$ on 11 vertices are shown in Figure 1. It should be noted that the 4 graphs with the highest edge counts shown in Figure 1 each have a single disconnected vertex. If that vertex is removed from each, the resulting graphs are all those with maximal $\forall vrn = 8$ on 10 vertices.

Recently, two new classes of graphs, called $RCC_{n,j}$ and $K_c \leftrightarrow^b K_c$ in [16, 15], have been shown to have high $\exists vrn$ if $n \geq 2$ and $j \geq 3$ or $c \geq 3$ and $2 \leq b \leq c-1$. $RCC_{n,j}$ graphs are regular graphs composed of n cycles with j vertices each, with edges between the cycles such that a vertex $v_{x,y}$ (cycle x, y-th vertex) is connected to $v_{x,(y+1 \bmod j)}$. For $G = RCC_{n,j}$, $|V(G)| = n \cdot j$, and $|E(G)| = n^2 \cdot j$. $K_c \leftrightarrow^b K_c$ graphs consist of two K_c components, connected with b extra edges, which share no common vertex. For $G = K_c \leftrightarrow^b K_c$, $|V(G)| = 2c$ and $|E(G)| = c(c-1)+b$. While it has been shown that these graphs have $\exists vrn > 3$, the specific value is only known for those graphs which were considered in [16, 15]. In Tables 4 and 5 we give the computed values of $\exists vrn$ and $\forall vrn$ of all such graphs with up to 16 vertices. These results contain two interesting patterns: $\exists vrn(RCC_{n,j}) = \forall vrn(RCC_{n,j})$, and $\forall vrn(K_c \leftrightarrow^b K_c) = c+2$.

graph	$\|V\|$	$\|E\|$	$\exists vrn$	$\forall vrn$
`C~`	4	2	4	4
`CR`	4	3	4	4
`Cr`	4	4	4	4
`E`?G`	6	3	4	4
`EwCW`	6	6	5	5
`E`dg`	6	7	4	5
`ETXW`	6	8	4	5
`Es\o`	6	9	5	5
`E}lw`	6	12	4	4
`G`?G?C`	8	4	4	4
`G?`H?c`	8	6	4	5
`GoCOZ?`	8	8	4	4
`G~?GW[`	8	12	6	6
`GoSsZc`	8	13	5	6
`G`iayw`	8	14	4	6
`G`rHpk`	8	14	4	6
`GTlai[`	8	15	5	6
`Gs`zro`	8	16	6	6
`G~rHx{`	8	20	4	4
`Gru~Z{`	8	22	4	5
`G~z\z{`	8	24	4	4

graph	$\|V\|$	$\|E\|$	$\exists vrn$	$\forall vrn$
`HwCW?CB`	9	9	5	5
`H}q|r|}`	9	27	5	5
`I`?G?C??G`	10	5	4	4
`Is`b?{]]?`	10	20	4	4
`I~{?GKF@w`	10	20	7	7
`Is@ipqF]G`	10	21	5	7
`IwC^FC]FW`	10	22	4	7
`IoSs[Xr[o`	10	22	5	7
`I`idA|]V_`	10	23	4	7
`I`rMXotKw`	10	23	5	7
`ITmzAdJPw`	10	24	5	7
`I~rH`cNBw`	10	25	4	4
`IsaBzx{^?`	10	25	7	7
`I~~v]}~~w`	10	40	4	4

Table 2: All graphs with $\exists vrn(G) > 3$ for $|V(G)| \leq 11$

<table>
<tr><th>graph</th><th>$|V|$</th><th>$|E|$</th><th>$\exists vrn$</th><th>$\forall vrn$</th></tr>
<tr><td>D?K</td><td>5</td><td>2</td><td>3</td><td>5</td></tr>
<tr><td>DAK</td><td>5</td><td>3</td><td>3</td><td>5</td></tr>
<tr><td>D@s</td><td>5</td><td>4</td><td>3</td><td>5</td></tr>
<tr><td>DBw</td><td>5</td><td>5</td><td>3</td><td>5</td></tr>
<tr><td>D`[</td><td>5</td><td>5</td><td>3</td><td>5</td></tr>
<tr><td>DJk</td><td>5</td><td>6</td><td>3</td><td>5</td></tr>
<tr><td>DR{</td><td>5</td><td>7</td><td>3</td><td>5</td></tr>
<tr><td>DN{</td><td>5</td><td>8</td><td>3</td><td>5</td></tr>
<tr><td>EAMw</td><td>6</td><td>7</td><td>3</td><td>6</td></tr>
<tr><td>EAlw</td><td>6</td><td>8</td><td>3</td><td>6</td></tr>
<tr><td>F?GYw</td><td>7</td><td>7</td><td>3</td><td>6</td></tr>
<tr><td>F?HXw</td><td>7</td><td>8</td><td>3</td><td>6</td></tr>
<tr><td>F?O|o</td><td>7</td><td>8</td><td>3</td><td>6</td></tr>
<tr><td>F@H]o</td><td>7</td><td>9</td><td>3</td><td>6</td></tr>
<tr><td>FPDIw</td><td>7</td><td>9</td><td>3</td><td>6</td></tr>
<tr><td>FKDhw</td><td>7</td><td>10</td><td>3</td><td>6</td></tr>
<tr><td>F@h^g</td><td>7</td><td>11</td><td>3</td><td>6</td></tr>
<tr><td>FANno</td><td>7</td><td>12</td><td>3</td><td>6</td></tr>
<tr><td>F`ozw</td><td>7</td><td>12</td><td>3</td><td>6</td></tr>
<tr><td>F@d~w</td><td>7</td><td>13</td><td>3</td><td>6</td></tr>
<tr><td>F`NZw</td><td>7</td><td>13</td><td>3</td><td>6</td></tr>
<tr><td>FAl~w</td><td>7</td><td>14</td><td>3</td><td>6</td></tr>
<tr><td>G?D@\g</td><td>8</td><td>9</td><td>3</td><td>7</td></tr>
<tr><td>G?CrUW</td><td>8</td><td>10</td><td>3</td><td>7</td></tr>
<tr><td>GhqXz{</td><td>8</td><td>18</td><td>3</td><td>7</td></tr>
<tr><td>GdW}z{</td><td>8</td><td>19</td><td>3</td><td>7</td></tr>
</table>

<table>
<tr><th>graph</th><th>$|V|$</th><th>$|E|$</th><th>$\exists vrn$</th><th>$\forall vrn$</th></tr>
<tr><td>H????CK</td><td>9</td><td>3</td><td>3</td><td>7</td></tr>
<tr><td>H????cK</td><td>9</td><td>4</td><td>3</td><td>7</td></tr>
<tr><td>H`?G?CB</td><td>9</td><td>6</td><td>3</td><td>7</td></tr>
<tr><td>HGCW?CB</td><td>9</td><td>7</td><td>3</td><td>7</td></tr>
<tr><td>H??H?lY</td><td>9</td><td>9</td><td>3</td><td>7</td></tr>
<tr><td>H??Gphe</td><td>9</td><td>10</td><td>3</td><td>7</td></tr>
<tr><td>H?GsIv~</td><td>9</td><td>17</td><td>3</td><td>7</td></tr>
<tr><td>H_MI\hr</td><td>9</td><td>17</td><td>3</td><td>7</td></tr>
<tr><td>H?LPMV~</td><td>9</td><td>18</td><td>3</td><td>7</td></tr>
<tr><td>H?luDly</td><td>9</td><td>18</td><td>3</td><td>7</td></tr>
<tr><td>HHDaW{~</td><td>9</td><td>18</td><td>3</td><td>7</td></tr>
<tr><td>H`MI\hr</td><td>9</td><td>18</td><td>3</td><td>7</td></tr>
<tr><td>HGSoz[~</td><td>9</td><td>19</td><td>3</td><td>7</td></tr>
<tr><td>HQopk|m</td><td>9</td><td>19</td><td>3</td><td>7</td></tr>
<tr><td>HhqXz~~</td><td>9</td><td>26</td><td>3</td><td>7</td></tr>
<tr><td>HdW}z~~</td><td>9</td><td>27</td><td>3</td><td>7</td></tr>
<tr><td>Hs\v~z~</td><td>9</td><td>29</td><td>3</td><td>7</td></tr>
<tr><td>HF~~vnn</td><td>9</td><td>30</td><td>3</td><td>7</td></tr>
<tr><td>Ht\~~~~</td><td>9</td><td>32</td><td>3</td><td>7</td></tr>
<tr><td>Hr^~~~~</td><td>9</td><td>33</td><td>3</td><td>7</td></tr>
<tr><td>I??Wvrex_</td><td>10</td><td>19</td><td>3</td><td>8</td></tr>
<tr><td>I?KpfbMr_</td><td>10</td><td>20</td><td>3</td><td>8</td></tr>
<tr><td>I`N}ECzMw</td><td>10</td><td>25</td><td>3</td><td>8</td></tr>
<tr><td>IxKWw~Fpw</td><td>10</td><td>26</td><td>3</td><td>8</td></tr>
<tr><td>J@?A?OTBdE?</td><td>11</td><td>13</td><td>3</td><td>8</td></tr>
<tr><td>J??_r?HDMK?</td><td>11</td><td>14</td><td>3</td><td>8</td></tr>
<tr><td>J???WX{[re?</td><td>11</td><td>19</td><td>3</td><td>8</td></tr>
<tr><td>J??WopwXrM?</td><td>11</td><td>20</td><td>3</td><td>8</td></tr>
<tr><td>JGCZz@`FWz_</td><td>11</td><td>25</td><td>3</td><td>8</td></tr>
<tr><td>JJCWW[NWzF_</td><td>11</td><td>26</td><td>3</td><td>8</td></tr>
<tr><td>J??Wvrexf~_</td><td>11</td><td>29</td><td>3</td><td>8</td></tr>
<tr><td>J?KpfbMrf~_</td><td>11</td><td>30</td><td>3</td><td>8</td></tr>
<tr><td>J`N}ECzM~~_</td><td>11</td><td>35</td><td>3</td><td>8</td></tr>
<tr><td>JxKWw~Fp~~_</td><td>11</td><td>36</td><td>3</td><td>8</td></tr>
<tr><td>Ju[|mnjZz~_</td><td>11</td><td>41</td><td>3</td><td>8</td></tr>
<tr><td>JyNS||}zz~_</td><td>11</td><td>42</td><td>3</td><td>8</td></tr>
</table>

Table 3: All graphs with maximal $\forall vrn(G)$ for $5 \leq |V(G)| \leq 11$

<table>
<tr><th colspan="2">graph</th><th>$|V|$</th><th>$|E|$</th><th>$\exists vrn$</th><th>$\forall vrn$</th></tr>
<tr><td>$RCC_{2,3}$</td><td>E}lw</td><td>6</td><td>12</td><td>4</td><td>4</td></tr>
<tr><td>$RCC_{2,4}$</td><td>Gs`zro</td><td>8</td><td>16</td><td>6</td><td>6</td></tr>
<tr><td>$RCC_{3,3}$</td><td>H}q|r|}</td><td>9</td><td>27</td><td>5</td><td>5</td></tr>
<tr><td>$RCC_{2,5}$</td><td>Is`b?{]]?</td><td>10</td><td>20</td><td>4</td><td>4</td></tr>
<tr><td>$RCC_{2,6}$</td><td>Ks_?BLeF_{N?</td><td>12</td><td>24</td><td>4</td><td>4</td></tr>
<tr><td>$RCC_{3,4}$</td><td>KsaCB|}^b{No</td><td>12</td><td>36</td><td>8</td><td>8</td></tr>
<tr><td>$RCC_{4,3}$</td><td>K}rD|y{^z~N{</td><td>12</td><td>48</td><td>6</td><td>6</td></tr>
<tr><td>$RCC_{2,7}$</td><td>Ms_??KEXbKBKEW]??</td><td>14</td><td>28</td><td>4</td><td>4</td></tr>
<tr><td>$RCC_{3,5}$</td><td>NsaCB`o[?^`}B{^_No?</td><td>15</td><td>45</td><td>5</td><td>5</td></tr>
<tr><td>$RCC_{5,3}$</td><td>N}rED}}nd{N~^}^}N~?</td><td>15</td><td>75</td><td>7</td><td>7</td></tr>
<tr><td>$RCC_{2,8}$</td><td>Os_??KE@_KKKWWEEBBBo?</td><td>16</td><td>32</td><td>4</td><td>4</td></tr>
<tr><td>$RCC_{4,4}$</td><td>OsaCCA?^z~N{^w^wN{B~?</td><td>16</td><td>64</td><td>10</td><td>10</td></tr>
</table>

Table 4: All $RCC_{n,j}$ graphs with $n \geq 2$ and $j \geq 3$ for $|V(G)| \leq 16$

<table>
<tr><th colspan="2">graph</th><th>$|V|$</th><th>$|E|$</th><th>$\exists vrn$</th><th>$\forall vrn$</th></tr>
<tr><td>$K_3 \leftrightarrow^2 K_3$</td><td>ETXW</td><td>6</td><td>8</td><td>4</td><td>5</td></tr>
<tr><td>$K_4 \leftrightarrow^2 K_4$</td><td>G`rHpk</td><td>8</td><td>14</td><td>4</td><td>6</td></tr>
<tr><td>$K_4 \leftrightarrow^3 K_4$</td><td>GTlai[</td><td>8</td><td>15</td><td>5</td><td>6</td></tr>
<tr><td>$K_5 \leftrightarrow^2 K_5$</td><td>IwC^FC]FW</td><td>10</td><td>22</td><td>4</td><td>7</td></tr>
<tr><td>$K_5 \leftrightarrow^3 K_5$</td><td>I`rMXotKw</td><td>10</td><td>23</td><td>5</td><td>7</td></tr>
<tr><td>$K_5 \leftrightarrow^4 K_5$</td><td>ITmzAdJPw</td><td>10</td><td>24</td><td>5</td><td>7</td></tr>
<tr><td>$K_6 \leftrightarrow^2 K_6$</td><td>K~?GW^o{G^@z</td><td>12</td><td>32</td><td>4</td><td>8</td></tr>
<tr><td>$K_6 \leftrightarrow^3 K_6$</td><td>KwC^FFbF_ybf</td><td>12</td><td>33</td><td>5</td><td>8</td></tr>
<tr><td>$K_6 \leftrightarrow^4 K_6$</td><td>K`rM][wLHdeN</td><td>12</td><td>34</td><td>5</td><td>8</td></tr>
<tr><td>$K_6 \leftrightarrow^5 K_6$</td><td>KTm|z@PQYJg^</td><td>12</td><td>35</td><td>6</td><td>8</td></tr>
<tr><td>$K_7 \leftrightarrow^2 K_7$</td><td>M~{?GKF@~o^@@}@|_</td><td>14</td><td>44</td><td>4</td><td>9</td></tr>
<tr><td>$K_7 \leftrightarrow^3 K_7$</td><td>M~?GW^o{N``{BtBr_</td><td>14</td><td>45</td><td>5</td><td>9</td></tr>
<tr><td>$K_7 \leftrightarrow^4 K_7$</td><td>MwC^FFbww{BhFJFF_</td><td>14</td><td>46</td><td>6</td><td>9</td></tr>
<tr><td>$K_7 \leftrightarrow^5 K_7$</td><td>M`rM]^NM@geRKVKN_</td><td>14</td><td>47</td><td>5</td><td>9</td></tr>
<tr><td>$K_7 \leftrightarrow^6 K_7$</td><td>MTm|||_SIPgfOnO^_</td><td>14</td><td>48</td><td>7</td><td>9</td></tr>
<tr><td>$K_8 \leftrightarrow^2 K_8$</td><td>O~~w?CB?wF_^~?~?_^oFz</td><td>16</td><td>58</td><td>4</td><td>10</td></tr>
<tr><td>$K_8 \leftrightarrow^3 K_8$</td><td>O~{?GKF@~o^@}B@}?}gNf</td><td>16</td><td>59</td><td>5</td><td>10</td></tr>
<tr><td>$K_8 \leftrightarrow^4 K_8$</td><td>O~?GW^o{N`}FBwBs`xW]N</td><td>16</td><td>60</td><td>6</td><td>10</td></tr>
<tr><td>$K_8 \leftrightarrow^5 K_8$</td><td>OwC^FFbw~FboFPFHbaww^</td><td>16</td><td>61</td><td>6</td><td>10</td></tr>
<tr><td>$K_8 \leftrightarrow^6 K_8$</td><td>O`rM]^NrxoE`KbKReDx_~</td><td>16</td><td>62</td><td>5</td><td>10</td></tr>
<tr><td>$K_8 \leftrightarrow^7 K_8$</td><td>OTm||}~WA_hBPFOfgJy@~</td><td>16</td><td>63</td><td>8</td><td>10</td></tr>
</table>

Table 5: All $K_c \leftrightarrow^b K_c$ graphs with $c \geq 3$ and $2 \leq b \leq c-1$ for $|V(G)| \leq 16$

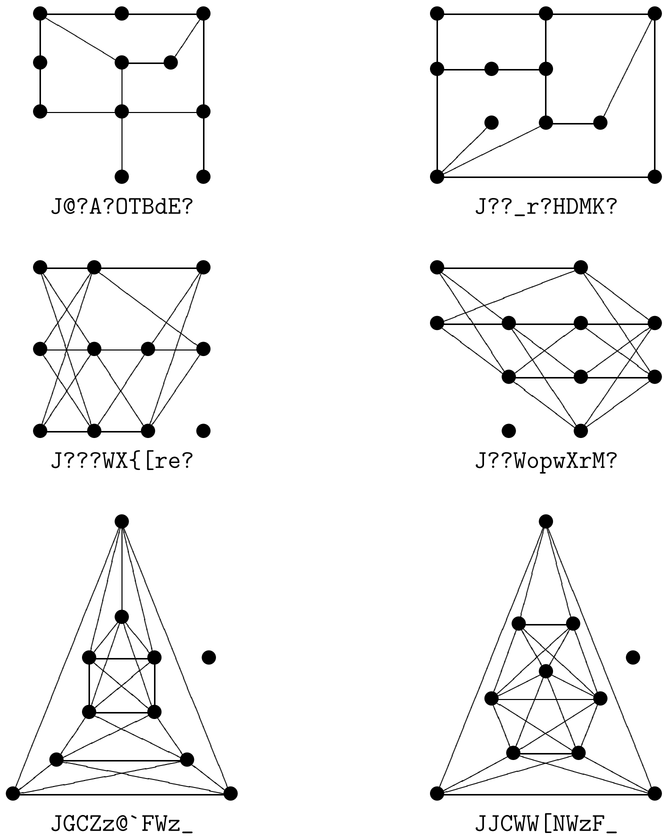


Figure 1: Graphs with maximal $\forall vrn(G) = 8$ on 11 vertices (first six in Table 3)

3 Edge Reconstruction Numbers

Table 6 shows the number of graphs with a given $\exists ern$ and $\forall ern$ for all orders between 3 and 11, inclusive, and at least one edge. Because there is exactly one graph of each order which has no edges, the number of graphs under consideration for edge-reconstruction is always one less than that of vertex-reconstruction.

For each listed $|V(G)| \geq 4$ there are 4 graphs which are not edge-reconstructible, although these graphs do not disprove the Edge Reconstruction Conjecture, as they all have less than four edges. These graphs fall into two pairs sharing the same $\mathcal{E}Deck$. Taking $n = |V(G)|$, the first pair is $P_3 \cup (n-3)K_1$ and $2K_2 \cup (n-4)K_1$, and the second pair is $K_3 \cup (n-3)K_1$ and $K_{1,3} \cup (n-4)K_1$. Similarly, there are 3 graphs for any order $n \geq 3$ which have $\exists ern = \forall ern = 1$: $K_2 \cup (n-2)K_1$, $\overline{K_2 \cup (n-2)K_1}$, and K_n.

We consider any edge-reconstructible graph with $\exists ern(G) \geq \frac{1}{2}|V(G)|$ to have

		graph order								
		3	4	5	6	7	8	9	10	11
unique graphs		3	10	33	155	1043	12345	274667	12005167	1018997863
not reconstructible		0	4	4	4	4	4	4	4	4
$\exists ern$	1	3	5	9	18	23	35	46	64	71
	2			14	115	980	12242	274523	12004951	1018997596
	3		1	6	16	31	57	81	130	167
	4				2	5	4	9	10	15
	5						3	3	5	6
	6							1	2	2
	7								1	1
	8									1
$\forall ern$	1	3	3	3	3	3	3	3	3	3
	2			2	14	19	51	152	1591	2479879
	3		3	8	28	131	1622	65814	5895154	748858136
	4			6	36	285	5059	141767	4976002	239960040
	5			8	46	394	3880	50196	925253	24213068
	6			2	15	128	952	10379	138350	2533007
	7				5	41	520	4171	47953	711284
	8				4	20	136	1228	11382	141498
	9					12	55	521	5704	67083
	10					4	26	202	1854	18352
	11					2	21	110	1070	9050
	12						8	57	359	2615
	13						4	37	292	2562
	14						2	10	68	512
	15						2	10	66	376
	16							2	23	188
	17								19	106
	18							2	8	30
	19								2	26
	20							2	2	10
	21								2	6
	22								2	8
	23								2	4
	24									2
	25									6
	26								2	2
	27									
	28									4
	29									
	30									
	31									
	32									
	33									2

Table 6: Counts of edge-reconstruction numbers by number of vertices

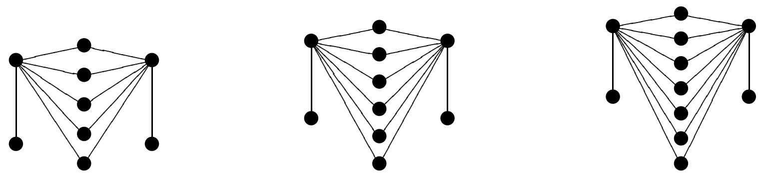
a high $\exists ern$, and list such graphs in Table 7. Many of the graphs in Table 7 are spanning subgraphs of $K_{r,2}$, including all graphs with $9 \leq |V(G)| \leq 11$. For graphs on between 9 and 11 vertices there is exactly one edge-reconstructible graph with the maximal $\exists ern$, and those graphs are shown in Figure 2. These graphs are part of the class of graphs which can be constructed by deleting two non-adjacent edges from a $K_{r,2}$. We note that all graphs in this class on between 5 and 11 vertices also have the properties that $\exists ern(G) = |V(G)| - 3$ and $\forall ern(G) = |V(G)| - 1$.

graph	$\|V\|$	$\|E\|$	$\exists ern$	$\forall ern$
Cr	4	4	3	3
DAK	5	3	3	3
DIK	5	4	3	3
DDW	5	4	3	4
DqK	5	5	3	3
DD[	5	5	3	5
DFw	5	6	3	3
E?D_	6	3	3	3
E?GW	6	3	3	3
E`?G	6	3	3	3
E@GW	6	4	3	3
E?Dg	6	4	3	4
E?F_	6	4	3	4
E?So	6	4	4	4
E?d_	6	4	4	4
EIGW	6	5	3	3
E?dg	6	5	3	4
E?Sw	6	5	3	5
E?\o	6	6	3	3
ECSw	6	6	3	3
EGDw	6	6	3	3
EoSo	6	6	3	3
E?lo	6	6	3	5
EElw	6	9	3	3
E}lw	6	12	3	3

graph	$\|V\|$	$\|E\|$	$\exists ern$	$\forall ern$
F??go	7	4	4	4
F?@H_	7	4	4	4
FJ?GW	7	6	4	4
F?@Xo	7	6	4	5
F?Azo	7	8	4	6
G???gW	8	4	4	4
G??@Go	8	4	4	4
G@K?GK	8	6	4	4
G??@Ww	8	6	4	5
G??CZ_	8	6	5	5
G??Axw	8	8	5	6
G??Dzw	8	10	5	7
H???AKw	9	6	5	5
H???@[]	9	8	5	6
H???A{}	9	10	5	7
H???D|}	9	12	6	8
I?????bF?	10	6	5	5
I?????VBo	10	8	5	6
I????AF]?	10	8	6	6
I?????nFo	10	10	5	7
I????@NLo	10	10	5	7
I????A^Zo	10	12	5	8
I????@^No	10	12	6	8
I????A^^o	10	14	7	9
J??????Oxw?	11	8	6	6
J??????Jw~?	11	12	6	8
J??????Vx~?	11	14	7	9
J??????nz~?	11	16	8	10

Table 7: All graphs with $\exists ern(G) \geq \frac{1}{2}|V(G)|$ for $|V(G)| \leq 11$

Table 8 lists those graphs with maximal $\forall ern$ for each order. It is interesting to note that in all cases the complements of the non-edge-reconstructible pair $P_3 \cup (n-3)K_1$ and $2K_2 \cup (n-4)K_1$ have a maximal $\forall ern$.



H???D|} $\exists ern = 6 \quad \forall ern = 8$

I????A~^o $\exists ern = 7 \quad \forall ern = 9$

J??????nz~? $\exists ern = 8 \quad \forall ern = 10$

Figure 2: The graphs with maximal $\exists ern(G)$ for $9 \leq |V(G)| \leq 11$

graph		$\|V\|$	$\|E\|$	$\exists ern$	$\forall ern$
P_4, $\overline{P_4}$	CR	4	3	1	3
$\overline{P_3 \cup K_1}$	CN	4	4	1	3
C_4, $\overline{2K_2}$	Cr	4	4	3	3
$\overline{P_3 \cup 2K_1}$	DN{	5	8	1	6
$\overline{2K_2 \cup 1K_1}$	Dr{	5	8	2	6
$\overline{P_4 \cup 2K_1}$	ER~w	6	12	1	8
$\overline{P_3 \cup K_2 \cup K_1}$	Er^w	6	12	1	8
$\overline{P_3 \cup 3K_1}$	EN~w	6	13	1	8
$\overline{2K_2 \cup 2K_1}$	Er~w	6	13	1	8
$\overline{P_3 \cup 4K_1}$	FN~~w	7	19	1	11
$\overline{2K_2 \cup 3K_1}$	Fr~~w	7	19	1	11
$\overline{P_3 \cup 5K_1}$	GN~~~{	8	26	1	15
$\overline{2K_2 \cup 4K_1}$	Gr~~~{	8	26	1	15
$\overline{P_3 \cup 6K_1}$	HN~~~~~	9	34	1	20
$\overline{2K_2 \cup 5K_1}$	Hr~~~~~	9	34	1	20
$\overline{P_3 \cup 7K_1}$	IN~~~~~~w	10	43	1	26
$\overline{2K_2 \cup 6K_1}$	Ir~~~~~~w	10	43	1	26
$\overline{P_3 \cup 8K_1}$	JN~~~~~~~~_	11	53	1	33
$\overline{2K_2 \cup 7K_1}$	Jr~~~~~~~~_	11	53	1	33

Table 8: All graphs with maximal $\forall ern(G)$ for $4 \leq |V(G)| \leq 11$

4 Algorithm and Computations

All the results presented in previous sections were obtained by use of special computer algorithms developed for this purpose, as well as the *nauty* package [14]. After introducing some notation on multisets (of graphs), this section describes the algorithms and their performance.

Definition 4. $m(S;\, x)$ is the multiplicity of an element x in a multiset $\mathcal{S}$ (the number of times x appears in $\mathcal{S}$).

Definition 5. $|\mathcal{S}| = \sum_{x \in \mathcal{S}} m(\mathcal{S};\, x)$ is the cardinality of a multiset $\mathcal{S}$.

Definition 6. $\mathbb{B}(\mathcal{S};\, m) = \{x \mid m(S;\, x) \geq m\}$ is the set of elements in $\mathcal{S}$ with multiplicity at least m. If m is omitted, then it is presumed to be 1, giving the basis set of $\mathcal{S}$.

The intersection ($\bigcap$) and union ($\bigcup$) of multisets preserves the minimal and maximal multiplicity of matching elements, while the additive union ($\biguplus$) sums the multiplicities of matching elements. Thus we have:

- $m(\mathcal{S}_1 \bigcap \mathcal{S}_2;\, x) = \min(m(\mathcal{S}_1;\, x), m(\mathcal{S}_2;\, x))$
- $m(\mathcal{S}_1 \bigcup \mathcal{S}_2;\, x) = \max(m(\mathcal{S}_1;\, x), m(\mathcal{S}_2;\, x))$
- $m(\mathcal{S}_1 \biguplus \mathcal{S}_2;\, x) = m(\mathcal{S}_1;\, x) + m(\mathcal{S}_2;\, x)$

In the following, a set will be considered to be a special case of multiset, where the multiplicity of all elements is one.

To determine both universal and existential reconstruction numbers the same primitive question is asked: "can a given subdeck $\mathcal{S}$ reconstruct G?" In order for $\mathcal{S}$ to not reconstruct G there must be another graph H which also has $\mathcal{S}$ as a subdeck. Therefore, in order to answer the question, either an example of a graph which shares the same subdeck must be found, or it must be proven that no such graph exists. We answer that question by computational search.

In order to narrow down the search space of graphs which may share a given subdeck, only graphs which share at least one card with G are considered. An expedient way of computing that search space is to perform the inverse operation to *Deck* (*$\mathcal{E}$Deck*) for each $C \in \mathit{Deck}(G)$ ($\mathcal{E}\mathit{Deck}(G)$).

Definition 7. *Extensions*(F) is the set of non-isomorphic graphs that results from adding one vertex to the graph F, and adding edges incident to the new vertex in every possible way.

Definition 8. *$\mathcal{E}$Extensions*(F) is the set of non-isomorphic graphs that results from adding one edge to the graph F in every possible way.

The following algorithm, inspired by that of by Brian McMullen [16, 15], was used to compute the reconstruction results in this work:

1. $\mathcal{D}_G \leftarrow \mathit{Deck}(G)$
2. for each $C \in \mathcal{D}_G$:
   (a) $\mathcal{H}_C \leftarrow \mathit{Extensions}(C) - G$
   (b) for each $H \in \mathcal{H}_C$ let $m(\mathcal{H}_C;\, H) \leftarrow \min(\ m(\mathit{Deck}(H);\, C),\ m(\mathcal{D}_G;\, C)\ )$
3. $\mathcal{H} \leftarrow \biguplus_{C \in \mathcal{D}_G} \mathcal{H}_C$
4. let $\forall rn(G) \leftarrow 1 + \max(m(\mathcal{H};\, H) :\ H \in \mathcal{H})$
5. let $\exists rn(G) \leftarrow \min(\ |\mathcal{S}| :\ (\mathcal{S} \subseteq \mathcal{D}_G) \wedge (\bigcap_{C \in \mathcal{S}} \mathbb{B}(\mathcal{H}_C;\, m(\mathcal{S};\, C)) = \emptyset)\ )$

To compute edge-deletion reconstruction numbers the same algorithm was used, with *Deck*/*Extensions* replaced by $\mathcal{E}$*Deck*/$\mathcal{E}$*Extensions*.

It is important to note that since isomorphic graphs are considered equivalent, a common implicit operation in this algorithm is the test of isomorphism. That is accomplished by use of the canonical labeling function in Brendan McKay's *nauty* [14] package. Each graph is canonically labeled as it is generated, and thereafter is simply tested for equality with others.

As canonical labeling itself is an expensive operation, it is beneficial to reduce the number of times it must be performed. The structural differences with the algorithm used in [15] and [16] are designed to reduce the number of canonical labelings that are required. By taking advantage of the fact that $H \in \mathit{Extensions}(C) \implies C \in \mathit{Deck}(H)$, we can see that $m(\mathit{Deck}(G);\, C) = 1 \implies m(\mathcal{H}_c;\, H) = 1$ in step 2b without performing any further calculations. This is especially useful as the probability that $m(\mathit{Deck}(G);\, C) = 1$ is quite high even for graphs of moderate order, as shown in Table 4. To further optimize cases where $m(\mathit{Deck}(G);\, C) > 1$, it can be noted that computing $m(\mathit{Deck}(H);\, C)$ only requires the inspection of those graphs in $\mathit{Deck}(H)$ that have the same number of edges as C.

	graph order								
	3	4	5	6	7	8	9	10	11
$P(m(D(G);\, C) = 1)$	.333	.300	.433	.511	.661	.789	.890	.947	.975
$P(m(\mathcal{E}D(G);\, C) = 1)$	.333	.357	.365	.466	.585	.727	.843	.917	.958

Table 9: Probability that any given card in a deck is unique

The benefit of these algorithmic advances is both a theoretical and measured speedup of rapidly approaching $|V(G)|$ as graph size increases, which enabled us to compute the results for 11 vertices. Table 10 lists the CPU time required to obtain the results presented in this paper. It should be noted that for the vertex reconstruction results only graphs on $\leq \frac{1}{2}\binom{|V(G)|}{2}$ edges were directly computed, as $vrn(G) = vrn(\overline{G})$ [8]. This does not apply, however, to the edge-reconstruction computations.

The results were primarily computed using AMD Opteron 248 CPUs, although results on 11 vertices were computed using the Center for Advancing the Study of Cyberinfrastructure (CASCI) compute cluster at the Rochester Institute of Technology (RIT), which consists of 94 1.4GHz Intel Pentium IV processors. In all runtimes presented below, times computed on the CASCI cluster are normalized to the equivalent of the Opteron 248 performance.

These values allow us to extrapolate the approximate CPU time which would be required to compute results for all 165091172592 graphs on 12 vertices. We estimate that approximately 200 equivalent CPU years would be needed to compute $\exists vrn$ and $\forall vrn$, and approximately 12 equivalent CPU years to compute $\exists ern$ and $\forall ern$. While this would seem to be within the reach of the largest compute clusters available today, goals beyond 12 vertices would seem to be out of reach for years to come with the algorithm used in this paper. Without a vast increase in computational power, we expect that the computation of reconstruction numbers for all graphs of order 13 will

			vertex reconstruction		edge reconstruction	
$\lvert V\rvert$	total graphs	$\le \frac{1}{2}\binom{\lvert V\rvert}{2}$ edges	total CPU time	ms per graph	total CPU time	ms per graph
6	156	78	0.02 seconds	0.26	0.04 seconds	0.25
7	1044	522	0.52 seconds	1.07	0.74 seconds	0.71
8	12346	6996	16.8 seconds	2.45	16.3 seconds	1.32
9	274668	154354	14.0 minutes	5.52	10.0 minutes	2.18
10	12005168	6002584	20.9 hours	12.6	10.7 hours	3.21
11	1018997864	509498932	174 days	29.5	23.7 days	4.02

Table 10: CPU time used to compute results

require a fundamental improvement in either the reconstruction number algorithm itself, or in the canonical labeling algorithm it so heavily relies upon.

We hope that the information presented in this paper will contribute insight into reconstruction numbers, and to the better understanding of the graph and edge reconstruction conjectures as a whole.

References

[1] K. Asciak, M. Francalanza, J. Lauri and W. Myrvold, A survey of some open questions in reconstruction numbers, Preprint, 2006.

[2] K. Asciak and J. Lauri, On disconnected graph with large reconstruction number, *Ars Combin.* 62 (2002), 173–181.

[3] L. W. Beineke, Families of graphs and digraphs, In Gross and Yellen [6], Chapter 1.2, pp. 20–28.

[4] B. Bollobás, Almost every graph has reconstruction number three, *J. Graph Theory* 14(1) (1990), 1–4.

[5] J. L. Gross and J. Yellen, Fundamentals of graph theory, In *Handbook of Graph Theory* [6], Chapter 1.1, pp. 2–19.

[6] J. L. Gross and J. Yellen, eds., *Handbook of Graph Theory*, CRC Press, Boca Raton, 2004.

[7] F. Harary, On the reconstruction of a graph from a collection of subgraphs, In *Theory of Graphs and its Applicns., Proc. Sympos. Smolenice*, 1963), pp. 47–52. Publ. House Czechoslovak Acad. Sci., Prague, 1964.

[8] F. Harary and M. Plantholt, The graph reconstruction number, *J. Graph Theory* 9(4) (1985), 451–454.

[9] P. J. Kelly, *On Isometric Transformations*, PhD thesis, The University of Wisconsin – Madison, 1942.

[10] J. Lauri, Vertex-deleted and edge-deleted subgraphs, In R. Ellul-Micallef and S. Fiorini, eds., *Collected Papers. Collegium Melitense Quatercentenary Celebrations (1592–1992)*. University of Malta, 1992. `http://staff.um.edu.mt/jlau/research/research.html`.

[11] J. Lauri, The reconstruction problem, In Gross and Yellen [6], Chapter 2.3, pp. 79–98.

[12] J. Lauri and R. Scapellato, *Topics in Graph Automorphisms and Reconstruction*, Cambridge University Press, Cambridge, UK, May 2003.

[13] B. D. McKay, Small graphs are reconstructible, *Australas. J. Combin.* 15 (1997), 123–126.

[14] B. D. McKay, *nauty User's Guide Version 2.2*, Comp. Sci. Dept., ANU, 2004, `http://cs.anu.edu.au/~bdm/nauty`.

[15] B. McMullen, Graph Reconstruction Numbers, Master's project, Rochester Inst. Tech., Rochester, NY 14623-5608, June 2005. `http://www.cs.rit.edu:8080/ms/static/spr/2004/4/bmm3056/`.

[16] B. McMullen and S. P. Radziszowski, Graph reconstruction numbers, *J. Combin. Math. Combin. Comput.* 62 (2007), 85–96. Errata: *ibid.* 63 (2007), 93–95.

[17] R. Molina, Correction of a proof on the ally-reconstruction number of a disconnected graph, *Ars Combin.* 40 (1995), 59–64.

[18] W. Myrvold, The ally-reconstruction number of a disconnected graph, *Ars Combin.* 28 (1989), 123–127.

[19] W. Myrvold, The ally-reconstruction number of a tree with five or more vertices is three, *J. Graph Theory* 14(2) (1990), 149–166.

[20] V. Nýdl, Graph reconstruction from subgraphs, *Discrete Math.* 235(1-3) (2001), 335–341.

(Received 17 Aug 2008; revised 31 May 2009)