Solution Algorithm to the Sam Loyd \((n^2 - 1)\) Puzzle

Kyle A. Bishop
Dustin L. Madsen

December 15, 2009
The Sam Loyd puzzle was a $4 \times 4$ grid invented in the 1870’s with numbers 0 through 15 on each of the grid tiles with the 15 tile being an open space.

Sam Loyd $4 \times 4$ Puzzle
The puzzle was scrambled using the empty space and was solved when the user brought it back to the original 1 through 15 configuration.

Start State $\rightarrow$ Goal State

The scrambled puzzle is called the start state and the solution puzzle is called the goal state.
Solvability

When attempting any Sam Loyd puzzle, solvability must first be checked in order to determine if the puzzle has a solution or not.

This is done by first checking the amount of permutations to get from the start state of the puzzle to the goal state.

Second, the distance the open space is from its goal state position is found.

If the sum of these two numbers is even, the puzzle is solvable; odd, then the puzzle is unsolvable.
Permutations

- Each permutation is defined as the act of replacing any tile in a given position for that position’s goal state tile.

- When actually playing the puzzle, each individual piece can only be exchanged with the empty space to achieve movement towards the goal state. When determining the evenness of the board to check solvability, this restriction is lifted.
A permutation cycle begins by first moving the tile in the 0 position into its goal state position.

The tile which was moved off the board is then put into its goal state position and so on.

A permutation cycle is complete when there are no more tiles left off the board to be moved into its goal state position.

If there is no tile currently in a given position for the goal state tile to replace, this is not considered a permutation.

The next permutation cycle begins with the next position not holding its goal state tile.
Permutations Example

- Using a simple $3 \times 3$ puzzle, checking evenness can be shown. 8 will represent the open space.

<table>
<thead>
<tr>
<th>6</th>
<th>3</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>7</td>
</tr>
</tbody>
</table>

Start State

- It is easier to perform the permutations by writing the puzzle into a list form going by rows.
Rewriting the above start state into list form:

```
6 3 0 1 8 4 2 5 7
```
Permutations Example

For this example, the 6 is currently in the 0 position and will be the first piece moved.

Blue squares will represent tiles which are being moved to their goal state.
Red squares will represent tiles which are in their goal state position.
The 6 moves into the position held by the 2, and the 2 is the next tile to be placed into its goal state position.
Permutations Example

- The 2 moves into the position held by the 0, and the 0 is the next tile to be placed into its goal state position.
The 0 is placed into the empty tile at the 0 position which was originally occupied the 6. This will end the first permutation cycle. These permutations are recorded for use later.

1st cycle: [6, 2, 0]
Now that all pieces are back onto the board, the second permutation cycle will begin.

The next tile to be moved is the 3 held in the 1 position.
The 3 is put into its goal state position held by the 1 tile.
The 1 tile is placed into its goal state position which was held by the 3, ending the second permutation cycle.

1st cycle: \([6, 2, 0]\)
2nd cycle: \([3, 1]\)
Now that all pieces are back onto the board, the third permutation cycle will begin.

The next tile to be moved is the 8 held in the 4 position.
Permutations Example

- The 8 replaces the 7.
Permutations Example

The 7 replaces the 5.

0 1 2 3 4 6 5 8

7

0 1 2 3 4 6 7 8

5

Kyle A. Bishop Dustin L. Madsen
Solution Algorithm to the Sam Loyd \((n^2 - 1)\) Puzzle
Permutations Example

- The 5 replaces the 4.

Diagram:

```
0 1 2 3
\[\text{Empty}\] 4 6 7 8
```

```
0 1 2 3
5 6 7 8
4
```

Solution Algorithm to the Sam Loyd \((n^2 - 1)\) Puzzle
Permutations Example

- The 4 is then the final piece to be moved and goes into the empty space which was occupied by the 8 tile.
- This ends the third and final permutation cycle.
Now that the puzzle is in its goal state, the sum of permutations is found.

- 1st cycle: [ 6, 2, 0 ]
- 2nd cycle: [ 3, 1 ]
- 3rd cycle: [ 8, 7, 5, 4 ]
Recall that each permutation is the act of replacing one tile for another.

Because of this, the last tile in each cycle is not counted as a permutation as it did not replace a tile when being placed into its goal state position.

With this in mind, each cycle will have \( n - 1 \) permutations, where \( n \) is the number of tiles in the cycle.
Permutations Sum

Summing the permutations of each cycle:

- 1st cycle: \([ 6, 2, 0 ]\) → 2 Permutations
- 2nd cycle: \([ 3, 1 ]\) → 1 Permutations
- 3rd cycle: \([ 8, 7, 5, 4 ]\) → 3 Permutations

Sum = 6 Permutations
Now that the sum of permutations has been recorded, the distance of the open space from its goal state position is found.

We look at the original start state puzzle in order to determine this distance.

The distance is known as the Manhattan Distance.
The Manhattan distance is the amount of movements required to move the open space from its start state position into its goal state position, the bottom right corner.

4 × 4 Manhattan Distance Matrix
Comparing the original sample start state with a $3 \times 3$ Manhattan Distance matrix, we can see that the Manhattan Distance is 2.
Manhattan Distance

- The sum of permutations and Manhattan distance is 8.
- Because this sum is even, the example puzzle is solvable.
Solvable puzzles are solved using the Parberry divide and conquer algorithm.

This algorithm repeatedly reduces the $n \times n$ puzzle into an $(n - 1) \times (n - 1)$ puzzle.
Puzzle Solution Algorithm

To solve any $n \times n$ puzzle using the Parberry divide and conquer algorithm:

- **Step 1**: Place tiles 0 to $(n - 1)$ in top row.
- **Step 2**: Place tiles $n, (n \times 2), \ldots, [n \times (n - 1)]$ in leftmost column.
- **Step 3**: Solve for $(n - 1) \times (n - 1)$ puzzle.
- Once the puzzle is reduced to $2 \times 2$, the final four tiles are rotated into place.

A $4 \times 4$ example will be presented:
**Puzzle Solution Algorithm**

- **Step 1:** Place tiles 0 to \((n - 1)\) in top row.
- **Tiles:** 0, 1, 2, 3.

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>5</td>
<td>7</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>11</td>
<td>4</td>
<td>9</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td>14</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>3</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>4</td>
<td>11</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>13</td>
<td>10</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>14</td>
<td>5</td>
<td>9</td>
<td></td>
</tr>
</tbody>
</table>
Puzzle Solution Algorithm

- **Step 2:** Place tiles $n$, $(n \times 2)$, . . . $[n \times (n - 1)]$ in leftmost column.
- **Tiles:** 4, 8, 12

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>15</td>
<td>11</td>
<td>4</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>11</td>
<td>15</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>4</td>
<td>6</td>
<td>1</td>
</tr>
</tbody>
</table>

Solution Algorithm to the Sam Loyd $(n^2 - 1)$ Puzzle
Step 3: Solve for \((n - 1) \times (n - 1)\) puzzle.
Continuing the algorithm, $3 \times 3 \rightarrow 2 \times 2$:

```
<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>6</td>
<td>11</td>
<td>15</td>
</tr>
<tr>
<td>8</td>
<td>13</td>
<td>10</td>
<td>7</td>
</tr>
<tr>
<td>12</td>
<td>14</td>
<td>5</td>
<td>9</td>
</tr>
</tbody>
</table>
```

```
<table>
<thead>
<tr>
<th>0</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>8</td>
<td>9</td>
<td>14</td>
<td>10</td>
</tr>
<tr>
<td>12</td>
<td>13</td>
<td>15</td>
<td>11</td>
</tr>
</tbody>
</table>
```
The $2 \times 2$ puzzle then needs to be rotated into the final goal state positions:
2 × 2 puzzle is rotated into goal state positions:
In order to perform the Parberry divide and conquer algorithm, specific tile movements are required.

- Each tile movement will depend upon its current location relative to its goal state position.
- The tile being moved is called the target tile.
- The goal state position is called the home location.
Tile Movements

There are four cases to consider when performing movement of tiles:

- Case 1: Target tile is to the left of its home column.

- Case 2: Target tile is on its home column.

- Case 3: Target tile is to the right of its home column, underneath home right diagonal.

- Case 4: Target tile is to the right of its home column, on or above the home right diagonal.
Tile Movements

Solution Algorithm to the Sam Loyd \((n^2 - 1)\) Puzzle
Case 1: Target tile is to the left of its home column.

The target is placed onto the left diagonal and moved into its home position.
Case 2: Target tile is on its home column.

The target is moved up and along its home column.
Case 3: Target tile is to the right of its home column, underneath home right diagonal.

The target is placed onto the right diagonal beneath the home position and moved into its home position.
Case 4: Target tile is to the right of its home column, on or above the home right diagonal.

The target is placed onto the home diagonal and moved into its home position.
When filling in the last two tiles for every row and column, a different approach is taken.
The $n - 2$ tile is placed into the $n - 1$ position.
The $n - 1$ tile is placed below the $n - 2$ tile.
The tiles are then rotated as in the final step of the Parberry divide and conquer algorithm.
Special Board Configuration

- Sometimes when performing the solution to a puzzle this situation may arise.

```
0  1  3  2
```

- This can not be solved by the methods previously presented.
Hayes, Richard  *The Sam Loyd 15-Puzzle*