**Problem Solving Using Weighted, Interacting TRAMS**

By C.D. Jones

Where TRAMS are swappable heuristic algorithms for story development using analogies, essentially allowing a “plug-in play” nature. Multiple TRAMS can be swapped in and out and linked together to produce procedural stories. The TRAMS work off analogies and allow a machine learning system to take analogies and use them to construct stories, each TRAM having its own strategy for that process.

TRAMS are not an original idea of mine and I will be adding a reference to the end of this paper once it is complete.

However, the use of weighted, interacting TRAMS is novel By generating a dynamic (growing) graph consisting of TRAMS as nodes and weights along the edges of the graph based off of their success, the TRAM system of procedural generation should improve over time, thanks to the rewarding of successful TRAMS in appropriate situations.

An example TRAM graph is as below:

TRAM Heuristic 1

Success Rate of 0.25, Weight = 0.25

Success Rate of 0.75, Weight= 0.75

TRAM Heurisitc 2

**OR**

TRAM Heuristic 3

This graph structure sort of combines ANN data networks with the TRAM system. When TRAM paths in the graph are successful, their weights are “back flushed” and made more heavy on the path from the leaf node (last TRAM used) back to the root (first TRAM used, first action).

Each TRAM Heuristic may be unique or there may be a repetition of similar TRAM types. The algorithm may swap TRAMS in and out of the graph at any level based on the weights and success of the algorithm so far.