

1127-03-278

**Nathanael L. Ackerman** (nate@math.harvard.edu) and **Cameron E. Freer\***  
(cameron@remine.com). *On the computability of graph Turing machines.*

We consider graph Turing machines, a model of parallel computation on a graph, in which each vertex is only capable of performing one of a finite number of operations. This model of computation is a natural generalization of several well-studied notions of computation, including ordinary Turing machines, cellular automata, and parallel graph dynamical systems. We analyze the power of computations that can take place in this model, both in terms of the degrees of computability of the functions that can be computed, and the time and space resources needed to carry out these computations. We further show that properties of the underlying graph have significant consequences for the power of computation thereby obtained. In particular, we show that every arithmetically definable set can be computed by a graph Turing machine in constant time, and that every computably enumerable Turing degree can be computed in constant time and linear space by a graph Turing machine whose underlying graph has finite degree. (Received February 06, 2017)