

# Disjointness Analysis for Java-Like Languages

James C. Jenista, Brian Demsky  
University of California, Irvine

Institute for Software Research  
UCI-ISR-09-1

February 2009

This paper presents a disjointness analysis for Java-like languages. Two objects are disjoint if the parts of the heap reachable from the two objects are disjoint. The analysis is based on static reachability graphs, which characterize the reachability of each object in the heap from a set of objects of interest. Reachability graphs contain nodes to represent objects and edges to represent heap references. The graphs are annotated with sets of reachability states that describe which objects can reach other objects. The analysis includes a global pruning step which analyzes the entire reachability graph to prune impossible reachability states that cannot be removed with local information alone.

We have developed an implementation of the analysis and have evaluated the implementation on several benchmarks. Our evaluation shows that the analysis reported all known aliases and no false aliases for our benchmark suite.