?
??
?
$Phantomization$
$STSCSTTS$
$PhantomizationRecoveryCleanUpPhantomization$
$Recovery$
$CleanUp$
$PhantomizationRecoveryCleanUpPhantomizationRecoveryCleanUp$
$Phantomization$
$Phantomization$
$Recovery$
$Cleanup$
$R1PhantomizationPhantomizationRecovery$
$??R1A12A$
$rootsXreachableXroots$
$PhantomizeRecoverCleanUpPhantomizationRecoveryRecoveryCleanUpPhantomizeRecoverCleanUp$
$G\alpha\alpha = \{A, B\}\alpha = \{A\}GG_AAG = G_A \cup G_BB\Gamma_B$

$\Gamma_BG_A \cap G_BG_A \cap G_BRecoveryG_A \cap G_BACleanUp\Gamma_B$
BG_AG_B
Theorem 1 (Cycle Invariant) *No strong cycles are possible, and all cycles formed in the graph should have at least one weak or p*
EST
$ESCSES$
$ETS \neq TT$
$ETST$
E
ET
ES
Theorem 2 (Termination) *Any mutations to a stable graph G will take $O(N)$ time steps to form a new stable graph G', where N*
$PhantomizationG$
$ABGG_AG_BG_A \cap G_B$

$\pi_A\pi_BAB$

$PhantomizationPhantomization\pi_BG_BG_BPhantomizationO(N)NG_B$
$RecoveryG_BPhantomizationRecoveredRecoveryO(N)$
$RecoveryCleanUp$
Theorem 3 (Safety) *Every node collected by our algorithm is indeed garbage and no nodes reachable by roots are collected.*
$RecoveryCleanup$
$V^CRoot \to ... \to V^A \to V^B \to ... \to V^CV^AV^BV^AV^BV^BV^AV^BRecoveryV^BV^C$
Theorem 4 (Liveness) *For a graph of finite size, our algorithm eventually collects all unreachable nodes.*
$liveall$
$Phantomization.PhantomizationRecoveryCleanup.$
$PhantomizationPhantomizationPhantomization$
Softw., Pract. Exper.
SIGPLAN Not.
ECOOP
ACM Trans. Program. Lang. Syst.
ACM Trans. Program. Lang. Syst.
Functional Programming Languages and Computer ArchitectureLecture Notes in Computer Science
Communications of the ACM
Software: Practice and Experience
Commun. ACM
Commun. ACM
Communications of the ACM
1981 ACM Symposium on Symbolic and Algebraic Computation
Garbage Collection and the Case for High-level Low- level Programming
Inf. Process. Lett.
The Computer Journal
SIGPLAN Not.
Garbage collection: algorithms for automatic dynamic memory management
ACM Trans. Program. Lang. Syst.
Inf. Process. Lett.
Inf. Process. Lett.
Inf. Process. Lett.
Commun. ACM
Commun. ACM
Commun. ACM
ACM Trans. Program. Lang. Syst.
A Cyclic Reference Counting Algorithm and Its Proof

SIGPLAN Not.
SIGPLAN Not.
ISMM
IPDPS
Comparative Performance Evaluation of Garbage Collection Algorithms